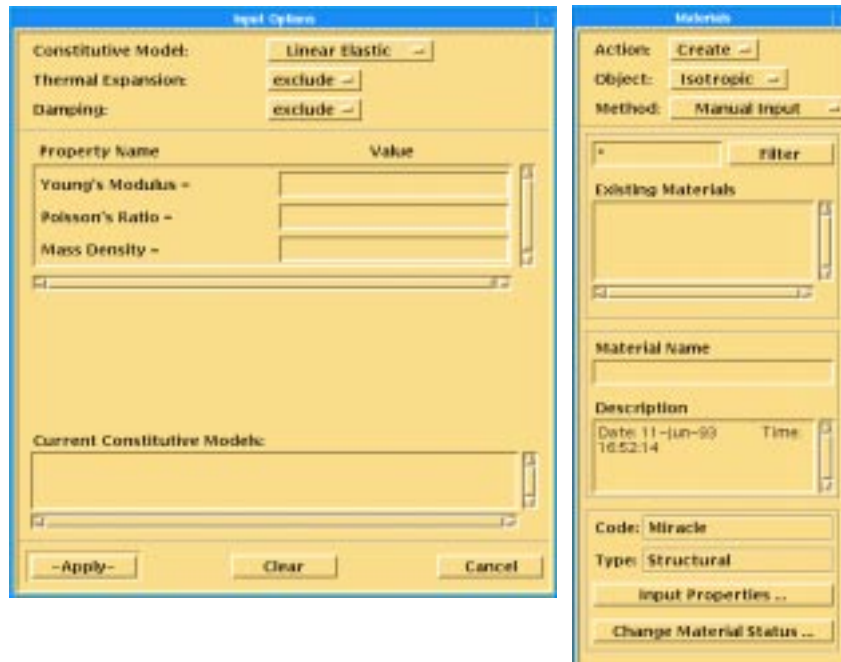


---

## EXERCISE 7

# *Loading Material Property Attributes*



### Objectives:

- Edit *exercise\_07.template* to add material options on the **Input Options** form.



**Problem Description:**

In this Exercise, you will modify *load\_Johncode()* file discussed in the class to change the arrangement of the material databoxes in the materials/input properties form. In particular, you will add two material option categories “magics”, and place items inside each one as shown in the figure below.

Property Name	Value
Reference Temperature -	
Young's Modulus -	
Poisson's Ratio -	
Mass Density -	
Thermal Expansion Coeff. -	
Structural Damping Coeff. -	

**load\_Johncode()  
(Before)**

**load\_miracle()  
(After)**

Property Name	Value
Young's Modulus -	
Poisson's Ratio -	
Mass Density -	

---

## Suggested Exercise Steps:

- Edit *exercise\_07.template* file and make modifications to add the new material property attributes, then rename the function to **exercise\_07.pcl**.
- Compile the PCL program. Be sure to use the C pre-processor.
- Open a new database using the *base.db* template.
- Run **exercise.07.pcl** and quit *p3*
- Verify the form.
- Verify the database relations/attributes.

## Exercise Procedure:

1. Either use vi or jot as the text editing tool. Open the file named *exercise\_07.template*. It should already exist in your directory. Make the appropriate modifications to the file to create the two “magics”.
2. Compile the function.

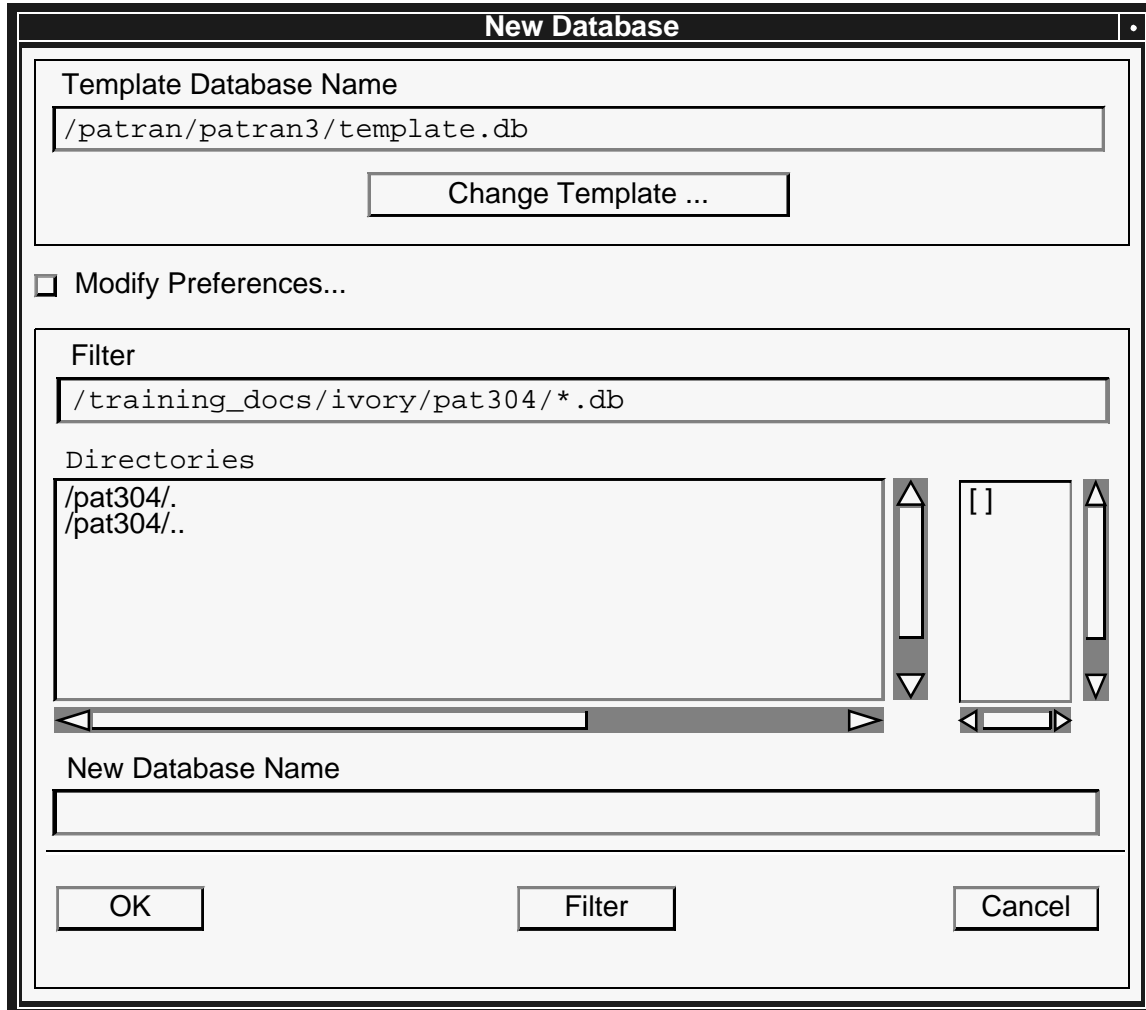
Type in the command line:

```
% /usr/lib/cpp -P exercise_07.pcl exercise_07.cpp  
% p3
```

Type p3 in your xterm. At the command line type:

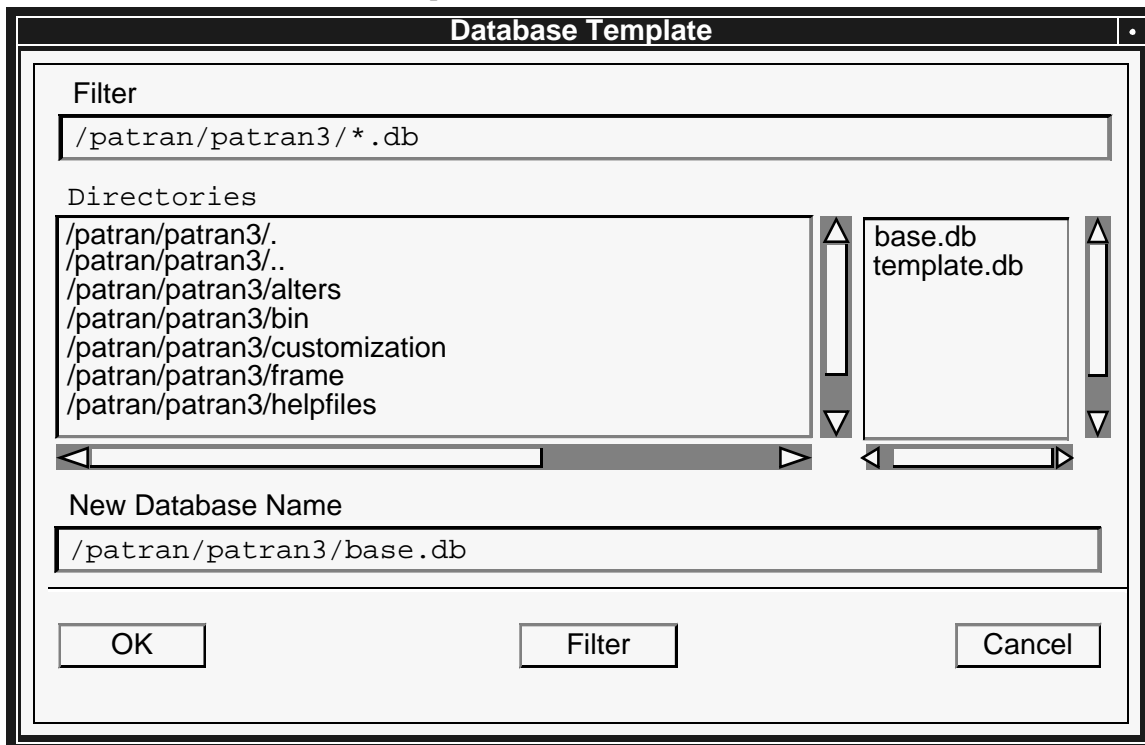
```
!!input exercise_07.cpp
```

In the *Control Panel*, select **File** from the *top menu bar*.  
Select **New Database...** from the pull down menu.



Click the **Change Template...** button.

In the *Database Template* form, select **base.db** from the Database Template List listbox.



Click the **OK** button.

EXERCISE 7

## Loading Material Property Attributes

In the *New Database* form, click in the New Database Name databox and enter the name **miracle\_template.db**.

The image shows a 'New Database' dialog box with the following fields and controls:

- Template Database Name:** A text box containing the path `/patran/patran3/base.db`. Below it is a button labeled 'Change Template ...'.
- Modify Preferences...:** A checkbox that is currently unchecked.
- Filter:** A text box containing the filter `/training_docs/ivory/pat304/*.db`.
- Directories:** A list box containing the entries `/pat304/` and `/pat304/..`. To the right of the list is a smaller empty list box with a single pair of square brackets `[]`.
- New Database Name:** A text box containing the name `miracle_template`.
- Buttons:** At the bottom are three buttons: 'OK', 'Filter', and 'Cancel'.

Click the **OK** button.

Now type **load\_miracle()** in the command window and hit return.

Close the database.

Quit p3.

3. Test your modifications to `load_miracle.pcl`

---

Begin PATRAN again by entering p3 at the prompt.

Click on the **File** in the *top menu bar*. Select **New Database...** from the pull-down menu. In the New Database form, change the Template Database Name to **miracle\_template.db**. Change the New Database Name to **exercise\_7.db**.

The image shows a screenshot of the "New Database" dialog box. The title bar reads "New Database". The dialog contains the following elements:

- Template Database Name:** A text input field containing "miracle\_template.db". Below it is a "Change Template ..." button.
- Modify Preferences...:** A checkbox that is currently unchecked.
- Filter:** A text input field containing the path and wildcard "/training\_docs/ivory/pat304/\*.res".
- Directories:** Two list boxes with scrollbars. The left list box contains the entries "/pat304/" and "/pat304/..". The right list box contains the entry "[miracle\_template.db]".
- New Database Name:** A text input field containing "exercise\_7.db".
- Buttons:** Three buttons are located at the bottom: "OK", "Filter", and "Cancel".

Click the **OK** button.

4. Verify the result.

Select the **Materials** radio button on the *Control Panel*:

You should see the following form appear:



---

Click the **Input Properties ...** button and the following form should appear:



Note that both the Thermal Expansion and Damping options are set to “include”. If you set the Thermal Expansion option to “exclude”, the Reference Temperature and Thermal Expansion Coeff. properties will disappear. If you set the Damping option to “exclude”, the Structural Damping Coeff. will disappear. If both are set to exclude, all three of the above properties should disappear, leaving only Young’s Modulus, Poisson’s Ratio, and Mass Density.

5. Using QLI, inspect the entries you have just created in the database. The following are some of the relations (i.e. Tables) you may want to examine.
- MATL\_MAGIC\_DEFINITION
  - allowable\_matl\_prop

To list all of the relations in the p3 database, use the following command

QLI> show relations

We have included a sample QLI session for your reference. However, feel free to let your curiosity take you to unfamiliar relations!

```
dresden_%qli
QLI> ready miracle_template.db
QLI> show relation MATL_MAGIC_DEFINITION
MATL_MAGIC_DEFINITION
ANALYSIS_CODE_ID short binary
ANALYSIS_TYPE_ID short binary
MAGIC_CATEGORY_ID long binary
MAGIC_LABEL varying text, length 32
Triggers defined for this relation:
DELETE_CASCADE_MATL_MAGIC_DEF Pre-erase, Sequence 0, Active

QLI> print MATL_MAGIC_DEFINITION

ANALYSIS ANALYSIS MAGIC
CODE TYPE CATEGORY MAGIC
ID ID ID LABEL
=====
1 1 20 Thermal Expansion
1 1 21 Damping

QLI> print allowable_matl_prop

ANALYSIS ANALYSIS MATL MATL CONSTIT
CODE TYPE CATEGORY PROP MODEL MAGIC MAGIC
ID ID ID ID ID NUMBER1 NUMBER2
=====

1 1 1 1 1 30 30 0 0 0 1
1 1 1 2 1 30 30 0 0 0 2
1 1 1 5 1 30 30 0 0 0 3
1 1 1 16 1 30 30 0 0 0 4
1 1 1 24 1 30 30 0 0 0 5
1 1 1 30 1 30 30 0 0 0 6
1 1 1 1 1 30 31 0 0 0 1
1 1 1 2 1 30 31 0 0 0 2
1 1 1 5 1 30 31 0 0 0 3
1 1 1 16 1 30 31 0 0 0 4
1 1 1 24 1 30 31 0 0 0 5
1 1 1 2 1 31 30 0 0 0 1
1 1 1 5 1 31 30 0 0 0 2
1 1 1 16 1 31 30 0 0 0 3
1 1 1 30 1 31 30 0 0 0 4
1 1 1 2 1 31 31 0 0 0 1
1 1 1 5 1 31 31 0 0 0 2
1 1 1 16 1 31 31 0 0 0 3
```

## Sample Solution

```

#include "app_ep_defn_ids.i"
#include "app_ep_prop_ids.i"
#include "mat_ind_cons.i"
#include "mat_words.p"
#include "elem_tests.p"

#define _YES 0
#define _NO 1
#define _NOT_VALID 0
#define _SCALAR 1
#define _VECTOR 2
#define _INTEGER 3
#define _STRING 4
#define _MATERIAL 5
#define _LIST 6
#define _FIELD 7
#define _NODAL 8
#define _COORD 9
#define _STRUCTURAL 1
#define _THERMAL 2

FUNCTION load_miracle()

    INTEGER status, bar_code, plate_code, solid_code, anal_code_id

    /*
    * Fetch the first unused analysis code id
    */

        •
        •
        •

    /*
    * Make the constitutive model valid
    */

    status = db_create_valid_const_models ( anal_code_id, @ /* Analysis code */
                                           _STRUCTURAL, @ /* Analysis type */
                                           MAT_CAT_ISOTROPIC, @ /* Material Category id */
                                           [1], @ /* List of material models */
                                           1 @ /* Number of material models */

    IF( status != 0 ) THEN
        write("Call to db_create_valid_const_models failed.")
        msg_to_form( status, 4, appcode(status), 1, 1., "" )
        RETURN
    END IF

    /*
    * Define material categories
    */

    status = db_create_mtl_magics_defn (***** 1 ***** @ /* Analysis code */
                                       ***** 1 ***** @ /* Analysis type */
                                       ***** 1 ***** @ /* Option category */
                                       "Thermal Expansion" ) /* Label for category */

    IF( status != 0 ) THEN
        write("Call to db_create_mtl_magics_defn failed.")
        msg_to_form( status, 4, appcode(status), 1, 1., "" )
        RETURN
    END IF

```

**EXERCISE 7*****Loading Material Property Attributes***

```
status = db_create_mtl_magics_defn ( ***** 1 ***** @ /* Analysis code */
                                ***** 1 ***** @ /* Analysis type */
                                ***** 1 ***** @ /* Option category */
                                "Damping" ) /* Label for category */

IF( status != 0 ) THEN
write("Call to db_create_mtl_magics_defn failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

/*
* define the items(options) which will be placed inside each of the options
*/

status = db_create_mtl_magics_options ( ***** 2 *****@ /* Analysis code */
                                       ***** 2 ***** @ /* Option (Add pg. 7-49) */
                                       ***** 2 ***** ) /* Label */

IF( status != 0 ) THEN
write("Call to db_create_mtl_magics_options failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_mtl_magics_options ( ***** 2 *****@ /* Analysis code */
                                       ***** 2 ***** @ /* Option (None pg. 7-49) */
                                       ***** 2 ***** ) /* Label */

IF( status != 0 ) THEN
write("Call to db_create_mtl_magics_options failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

/*
* Group/place the items inside each of the categories
*/

status = db_create_mtl_allowable_magics ( ***** 3 *****@ /* Analysis code */
                                         ***** 3 ***** @ /* Material Category */
                                         ***** 3 ***** @ /* Option Category ID */
                                         ***** 3 ***** @ /* List of material options */
                                         ***** 3 ***** ) /* Number of material options */

IF( status != 0 ) THEN
write("Call to db_create_mtl_allowable_magics failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_mtl_allowable_magics ( ***** 3 ***** @ /* Analysis code */
                                         ***** 3 ***** @ /* Material Category */
                                         ***** 3 ***** @ /* Option Category ID */
                                         ***** 3 ***** @ /* List of material options */
                                         ***** 3 ***** ) /* Number of material options */

IF( status != 0 ) THEN
write("Call to db_create_mtl_allowable_magics failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

/*
* Tie the categories to the material model (i.e. lines, non-linear ...)
*/
```

```

status = db_create_mtl_magics ( ***** 4 ***** @ /* Analysis code */
                             ***** 4 ***** @ /* Analysis type */
                             ***** 4 ***** @ /* Material Category */
                             ***** 4 ***** @ /* Material Constitutive Model */
                             ***** 4 ***** @ /* List of Option categories */
                             2 ) /* Number of Option categories */

/*
* Define material words
*/

status = db_create_matl_prop_alias ( anal_code_id, @ /* Analysis code */
                                   _STRUCTURAL, @ /* Analysis type */
                                   1, @ /* Material category */
                                   1, @ /* Material Constitutive model */
                                   _REF_TEMP, @ /* Material Word */
                                   "Reference Temperature", @ /* Label (Alias) */
                                   _NO_FIELD ) /* Type of field */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_matl_prop_alias ( anal_code_id, @ /* Analysis code */
                                   _STRUCTURAL, @ /* Analysis type */
                                   1, @ /* Material category */
                                   1, @ /* Material Constitutive model */
                                   _ELASTIC_MODULUS, @ /* Material Word */
                                   "Young's Modulus", @ /* Label (Alias) */
                                   _TEMP_DEP_FIELD ) /* Type of field */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_matl_prop_alias ( anal_code_id, @ /* Analysis code */
                                   _STRUCTURAL, @ /* Analysis type */
                                   1, @ /* Material category */
                                   1, @ /* Material Constitutive model */
                                   _POISSONS_RATIO, @ /* Material Word */
                                   "Poisson's Ratio", @ /* Label (Alias) */
                                   _TEMP_DEP_FIELD ) /* Type of field */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_matl_prop_alias ( anal_code_id, @ /* Analysis code */
                                   _STRUCTURAL, @ /* Analysis type */
                                   1, @ /* Material category */
                                   1, @ /* Material Constitutive model */
                                   _DENSITY, @ /* Material Word */
                                   "Mass Density", @ /* Label (Alias) */
                                   _TEMP_DEP_FIELD ) /* Type of field */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

```

**EXERCISE 7*****Loading Material Property Attributes***

```

status = db_create_matl_prop_alias ( anal_code_id,    @ /* Analysis code          */
                                   _STRUCTURAL,      @ /* Analysis type           */
                                   1,                @ /* Material category       */
                                   1,                @ /* Material Constitutive model */
                                   _THERMAL_EXPANSION, @ /* Material Word           */
                                   "Thermal Expansion Coeff.", @ /* Label (Alias)          */
                                   _TEMP_DEP_FIELD   ) /* Type of field           */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_matl_prop_alias ( anal_code_id,    @ /* Analysis code          */
                                   _STRUCTURAL,      @ /* Analysis type           */
                                   1,                @ /* Material category       */
                                   1,                @ /* Material Constitutive model */
                                   _STRUCTURAL_DAMPING, @ /* Material Word           */
                                   "Structural Damping Coeff.", @ /* Label (Alias)          */
                                   _TEMP_DEP_FIELD   ) /* Type of field           */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

/*
* Define the material records
*/

status = db_create_allowable_matl_prop ( ***** 5 *****@ /* Analysis code          */
                                       ***** 5 ***** @ /* Analysis type           */
                                       ***** 5 ***** @ /* Material Category       */
                                       ***** 5 ***** @ /* Material Constitutive model */
                                       ***** 5 ***** @ /* List of 5 option categories */
                                       @
                                       @
                                       @
                                       @
                                       @
                                       @
                                       @
                                       @ /* List of material words   */
                                       ***** 5 ***** ) /* Number of material words */

IF( status != 0 ) THEN
write("Call to db_create_allowable_matl_prop failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_allowable_matl_prop ( ***** 5 *****@ /* Analysis code          */
                                       ***** 5 ***** @ /* Analysis type           */
                                       ***** 5 ***** @ /* Material Category       */
                                       ***** 5 ***** @ /* Material Constitutive model */
                                       ***** 5 ***** @ /* List of 5 option categories */
                                       @
                                       @
                                       @
                                       @
                                       @
                                       @
                                       @
                                       @ /* List of material words   */
                                       ***** 5 ***** ) /* Number of material words */

IF( status != 0 ) THEN
write("Call to db_create_allowable_matl_prop failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

```

```
status = db_create_allowable_matl_prop ( ***** 5 *****@ /* Analysis code */
***** 5 ***** @ /* Analysis type */
***** 5 ***** @ /* Material Category */
***** 5 ***** @ /* Material Constitutive model */
***** 5 ***** @ /* List of 5 option categories */
***** 5 ***** @
***** 5 ***** @
***** 5 ***** @
***** 5 ***** @ /* List of material words */
***** 5 ***** ) /* Number of material words */

IF( status != 0 ) THEN
write("Call to db_create_allowable_matl_prop failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_allowable_matl_prop ( ***** 5 *****@ /* Analysis code */
***** 5 ***** @ /* Analysis type */
***** 5 ***** @ /* Material Category */
***** 5 ***** @ /* Material Constitutive model */
***** 5 ***** @ /* List of 5 option categories */
***** 5 ***** @
***** 5 ***** @
***** 5 ***** @ /* List of material words */
***** 5 ***** ) /* Number of material words */

IF( status != 0 ) THEN
write("Call to db_create_allowable_matl_prop failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF
$ Set the new Miracle to be the Default Analysis Code
uil_pref_analysis.set_analysis_pref( "Miracle", "Structural", ".inp", ".ans" )
$ End
END FUNCTION /* load_miracle */
```

## Solutions

```

1)status = db_create_mtl_magics_defn ( anal_code_id, @ /* Analysis code */
20, @ /* Option category */
/*Thermal Expansion*/ ) /* Label for category */
status = db_create_mtl_magics_defn ( anal_code_id, @ /* Analysis code */
21, @ /* Option category */
/*STRUCTURAL, @ /* Analysis type */
status = db_create_mtl_magics_options ( anal_code_id, @ /* Analysis code */
31, @ /* Option (Add pg. 7-49) */
/*exclude*/ /* Label */
status = db_create_mtl_magics_options ( anal_code_id, @ /* Analysis code */
30, @ /* Option (None pg. 7-49) */
/*include*/ /* Label */
3)status = db_create_mtl_allowable_magics ( anal_code_id, @ /* Analysis code */
20, @ /* Option Category ID */
[30,31], @ /* List of material options */
2 ) /* Number of material options */
status = db_create_mtl_allowable_magics ( anal_code_id, @ /* Analysis code */
MAT_CAT_ISOTROPIC, @ /* Material Category */
21, @ /* Option Category ID */
[30,31], @ /* List of material options */
2 ) /* Number of material options */
4)status = db_create_mtl_magics ( anal_code_id, @ /* Analysis code */
/*STRUCTURAL, @ /* Analysis type */
MAT_CAT_ISOTROPIC, @ /* Material Category */
1, @ /* Material Constitutive Model */
[20, 21 ], @ /* List of Option categories */
2 ) /* Number of Option categories */
5)status = db_create_allowable_mtl_prop ( anal_code_id, @ /* Analysis code */
/*STRUCTURAL, @ /* Analysis type */
1, @ /* Material Category */
1, @ /* Material Constitutive model */
[30,30,0,0], @ /* List of 5 option categories */
[REF_TEMP, @
ELASTIC MODULUS, @
POISSONS RATIO, @
DENSITY, @
THERMAL_EXPANSION, @
/*STRUCTURAL DAMPING], @ /* List of material words */
6 ) /* Number of material words */

```

```
status = db.create_allowable_matl_prop ( anal_code_id, @ /* Analysis code */
STRUC_TURAT, @ /* Analysis type */
1, @ /* Material Category */
1, @ /* Material Constitutive model */
[31,31,0,0], @ /* List of 5 option categories */
@
ELASTIC_MODULUS, @
POISSONS_RATIO, @
DENSITY, @
STRUC_TURAT_DAMPING, @ /* List of material words */
4 ) /* Number of material words */

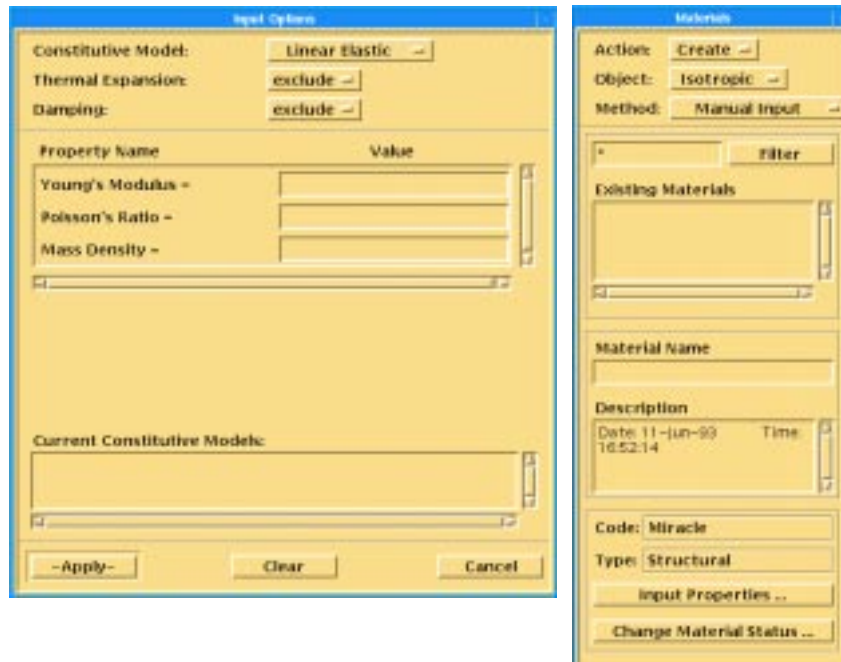
status = db.create_allowable_matl_prop ( anal_code_id, @ /* Analysis code */
STRUC_TURAT, @ /* Analysis type */
1, @ /* Material Category */
1, @ /* Material Constitutive model */
[31,30,0,0], @ /* List of 5 option categories */
@
ELASTIC_MODULUS, @
POISSONS_RATIO, @
DENSITY, @
THERMAL_EXPANSION, @ /* List of material words */
5 ) /* Number of material words */

status = db.create_allowable_matl_prop ( anal_code_id, @ /* Analysis code */
STRUC_TURAT, @ /* Analysis type */
1, @ /* Material Category */
1, @ /* Material Constitutive model */
[30,31,0,0], @ /* List of 5 option categories */
@
REF_TEMP, @
ELASTIC_MODULUS, @
POISSONS_RATIO, @
DENSITY, @
THERMAL_EXPANSION, @ /* List of material words */
5 ) /* Number of material words */
```

---

## EXERCISE 7

# *Loading Material Property Attributes*



### Objectives:

- Edit *exercise\_07.template* to add material options on the **Input Options** form.



**Problem Description:**

In this Exercise, you will modify *load\_Johncode()* file discussed in the class to change the arrangement of the material databoxes in the materials/input properties form. In particular, you will add two material option categories “magics”, and place items inside each one as shown in the figure below.

Property Name	Value
Reference Temperature -	
Young's Modulus -	
Poisson's Ratio -	
Mass Density -	
Thermal Expansion Coeff. -	
Structural Damping Coeff. -	

**load\_Johncode()  
(Before)**

**load\_miracle()  
(After)**

Property Name	Value
Young's Modulus -	
Poisson's Ratio -	
Mass Density -	

---

## Suggested Exercise Steps:

- Edit *exercise\_07.template* file and make modifications to add the new material property attributes, then rename the function to **exercise\_07.pcl**.
- Compile the PCL program. Be sure to use the C pre-processor.
- Open a new database using the *base.db* template.
- Run **exercise.07.pcl** and quit *p3*
- Verify the form.
- Verify the database relations/attributes.

## Exercise Procedure:

1. Either use vi or jot as the text editing tool. Open the file named *exercise\_07.template*. It should already exist in your directory. Make the appropriate modifications to the file to create the two “magics”.
2. Compile the function.

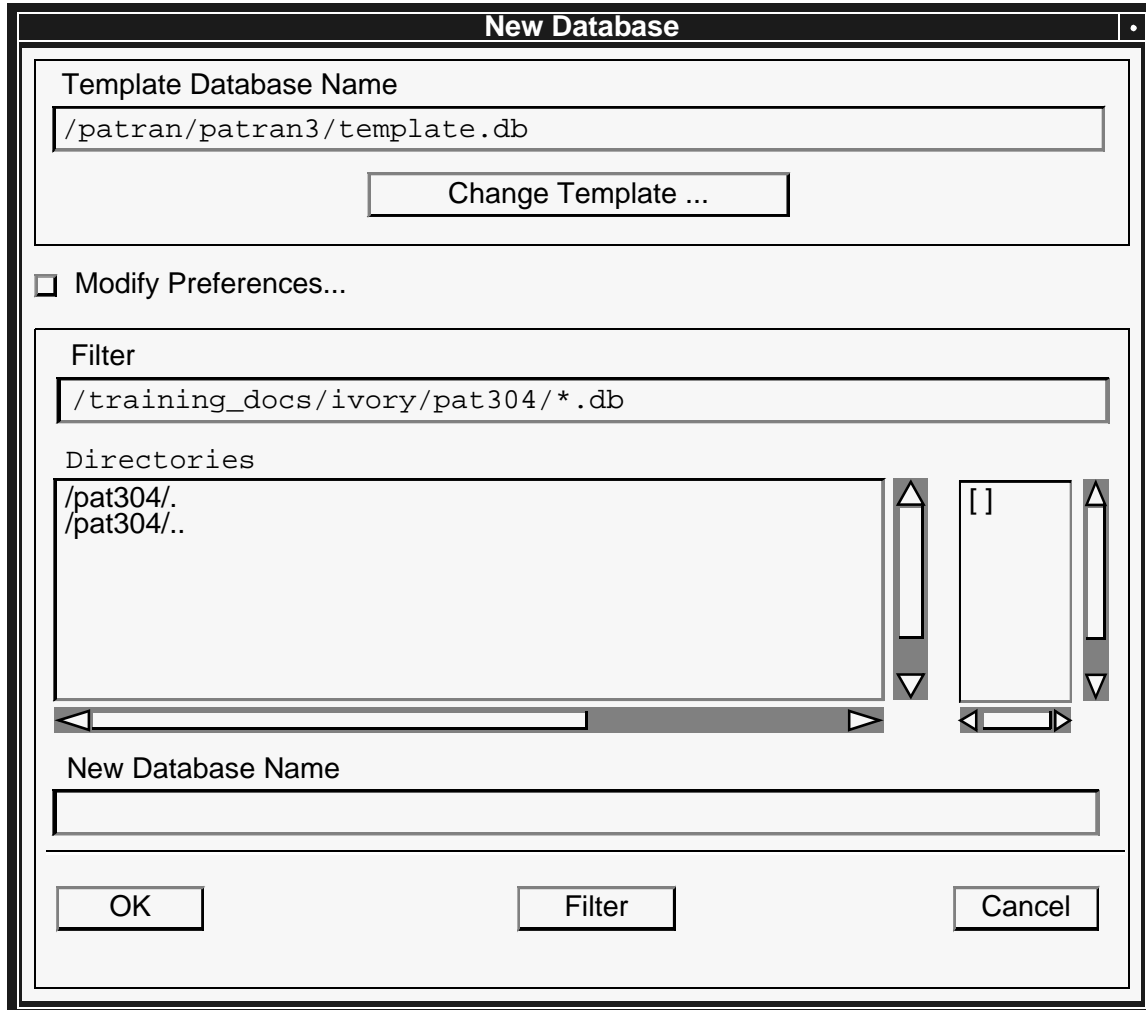
Type in the command line:

```
% /usr/lib/cpp -P exercise_07.pcl exercise_07.cpp  
% p3
```

Type p3 in your xterm. At the command line type:

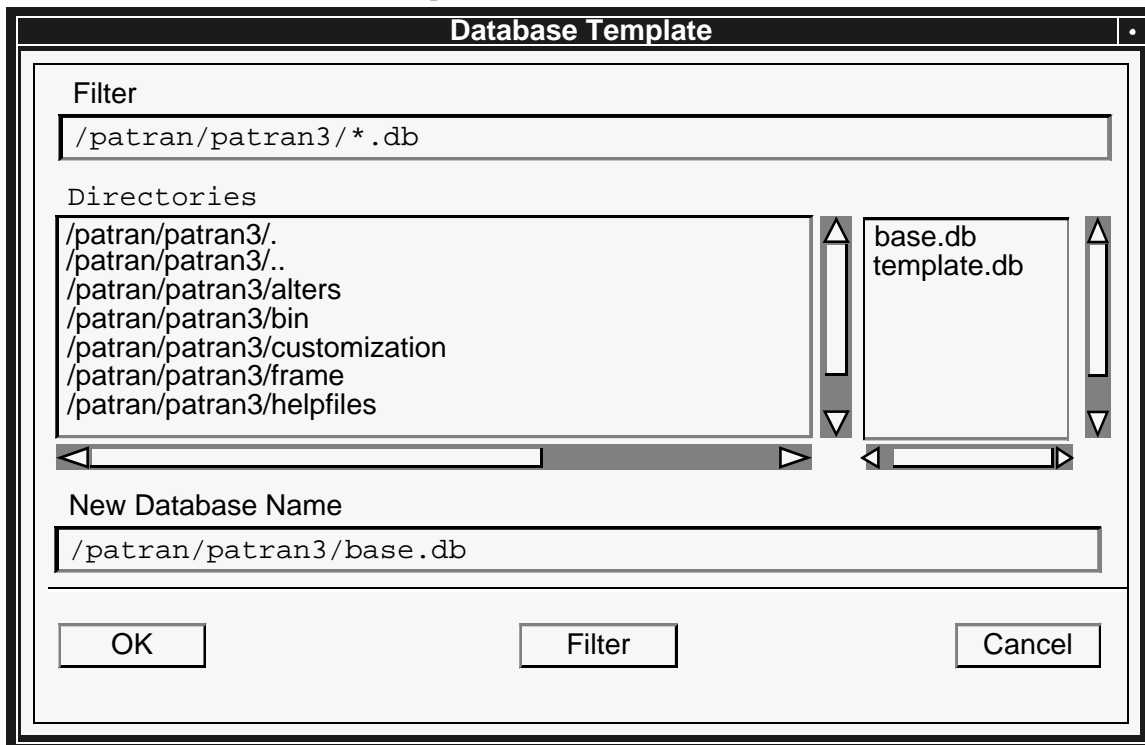
```
!!input exercise_07.cpp
```

In the *Control Panel*, select **File** from the *top menu bar*.  
Select **New Database...** from the pull down menu.



Click the **Change Template...** button.

In the *Database Template* form, select **base.db** from the Database Template List listbox.



Click the **OK** button.

EXERCISE 7

## Loading Material Property Attributes

In the *New Database* form, click in the New Database Name databox and enter the name **miracle\_template.db**.

The image shows a 'New Database' dialog box with the following fields and controls:

- Template Database Name:** A text box containing the path `/patran/patran3/base.db`. Below it is a button labeled 'Change Template ...'.
- Modify Preferences...:** An unchecked checkbox.
- Filter:** A text box containing the filter `/training_docs/ivory/pat304/*.db`.
- Directories:** A list box containing `/pat304/` and `/pat304/..`. To its right is an empty list box containing `[]`.
- New Database Name:** A text box containing the name `miracle_template`.
- Buttons:** 'OK', 'Filter', and 'Cancel' buttons are located at the bottom of the dialog.

Click the **OK** button.

Now type **load\_miracle()** in the command window and hit return.

Close the database.

Quit p3.

3. Test your modifications to `load_miracle.pcl`

---

Begin PATRAN again by entering p3 at the prompt.

Click on the **File** in the *top menu bar*. Select **New Database...** from the pull-down menu. In the New Database form, change the Template Database Name to **miracle\_template.db**. Change the New Database Name to **exercise\_7.db**.

The image shows a 'New Database' dialog box with the following fields and controls:

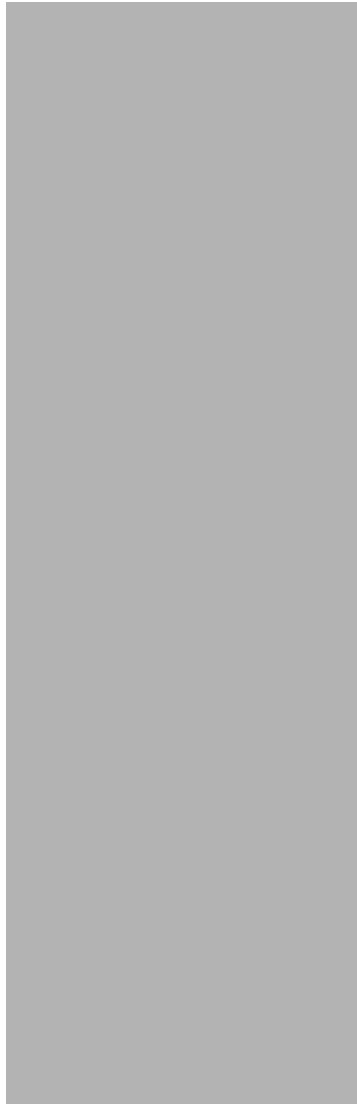
- Template Database Name:** Text field containing 'miracle\_template.db'. Below it is a 'Change Template ...' button.
- Modify Preferences...**
- Filter:** Text field containing '/training\_docs/ivory/pat304/\*.res'.
- Directories:** Two list boxes. The left one contains '/pat304/' and '/pat304/..'. The right one contains '[miracle\_template.db]'.
- New Database Name:** Text field containing 'exercise\_7.db'.
- Buttons:** 'OK', 'Filter', and 'Cancel' at the bottom.

Click the **OK** button.

4. Verify the result.

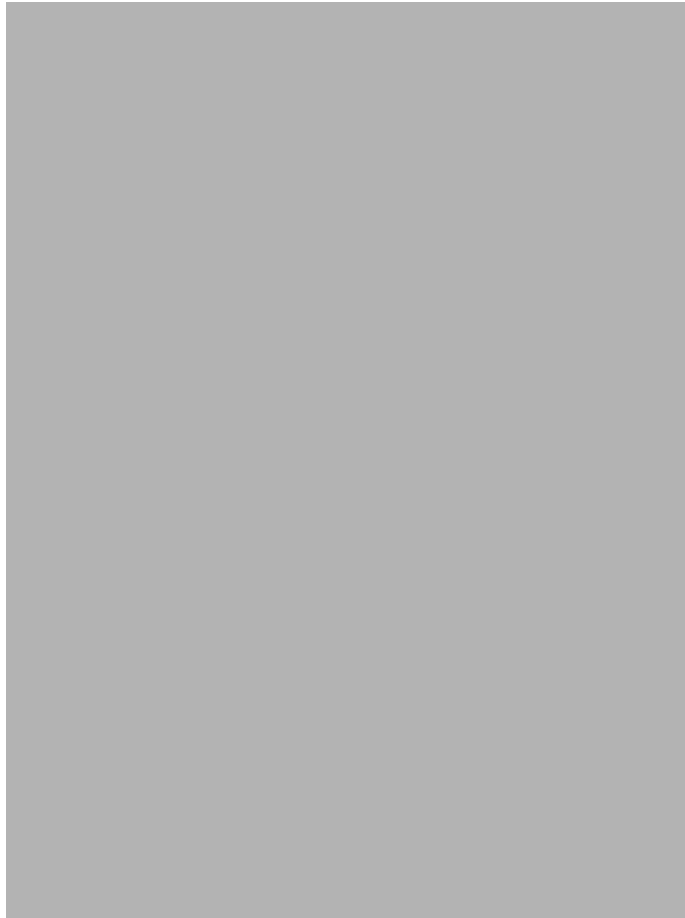
Select the **Materials** radio button on the *Control Panel*:

You should see the following form appear:



---

Click the **Input Properties ...** button and the following form should appear:



Note that both the Thermal Expansion and Damping options are set to “include”. If you set the Thermal Expansion option to “exclude”, the Reference Temperature and Thermal Expansion Coeff. properties will disappear. If you set the Damping option to “exclude”, the Structural Damping Coeff. will disappear. If both are set to exclude, all three of the above properties should disappear, leaving only Young’s Modulus, Poisson’s Ratio, and Mass Density.

5. Using QLI, inspect the entries you have just created in the database. The following are some of the relations (i.e. Tables) you may want to examine.
- MATL\_MAGIC\_DEFINITION
  - allowable\_matl\_prop

To list all of the relations in the p3 database, use the following command

QLI> show relations

We have included a sample QLI session for your reference. However, feel free to let your curiosity take you to unfamiliar relations!

```
dresden_%qli
QLI> ready miracle_template.db
QLI> show relation MATL_MAGIC_DEFINITION
MATL_MAGIC_DEFINITION
ANALYSIS_CODE_ID short binary
ANALYSIS_TYPE_ID short binary
MAGIC_CATEGORY_ID long binary
MAGIC_LABEL varying text, length 32
Triggers defined for this relation:
DELETE_CASCADE_MATL_MAGIC_DEF Pre-erase, Sequence 0, Active

QLI> print MATL_MAGIC_DEFINITION

ANALYSIS ANALYSIS MAGIC
CODE TYPE CATEGORY MAGIC
ID ID ID LABEL
=====
1 1 20 Thermal Expansion
1 1 21 Damping

QLI> print allowable_matl_prop

ANALYSIS ANALYSIS MATL MATL CONSTIT
CODE TYPE CATEGORY PROP MODEL MAGIC MAGIC
ID ID ID ID ID NUMBER1 NUMBER2
=====

1 1 1 1 1 30 30 0 0 0 1
1 1 1 2 1 30 30 0 0 0 2
1 1 1 5 1 30 30 0 0 0 3
1 1 1 16 1 30 30 0 0 0 4
1 1 1 24 1 30 30 0 0 0 5
1 1 1 30 1 30 30 0 0 0 6
1 1 1 1 1 30 31 0 0 0 1
1 1 1 2 1 30 31 0 0 0 2
1 1 1 5 1 30 31 0 0 0 3
1 1 1 16 1 30 31 0 0 0 4
1 1 1 24 1 30 31 0 0 0 5
1 1 1 2 1 31 30 0 0 0 1
1 1 1 5 1 31 30 0 0 0 2
1 1 1 16 1 31 30 0 0 0 3
1 1 1 30 1 31 30 0 0 0 4
1 1 1 2 1 31 31 0 0 0 1
1 1 1 5 1 31 31 0 0 0 2
1 1 1 16 1 31 31 0 0 0 3
```

## Sample Solution

```

#include "app_ep_defn_ids.i"
#include "app_ep_prop_ids.i"
#include "mat_ind_cons.i"
#include "mat_words.p"
#include "elem_tests.p"

#define _YES 0
#define _NO 1
#define _NOT_VALID 0
#define _SCALAR 1
#define _VECTOR 2
#define _INTEGER 3
#define _STRING 4
#define _MATERIAL 5
#define _LIST 6
#define _FIELD 7
#define _NODAL 8
#define _COORD 9
#define _STRUCTURAL 1
#define _THERMAL 2

FUNCTION load_miracle()

    INTEGER status, bar_code, plate_code, solid_code, anal_code_id

    /*
    * Fetch the first unused analysis code id
    */

        •
        •
        •

    /*
    * Make the constitutive model valid
    */

    status = db_create_valid_const_models ( anal_code_id, @ /* Analysis code */
                                           _STRUCTURAL, @ /* Analysis type */
                                           MAT_CAT_ISOTROPIC, @ /* Material Category id */
                                           [1], @ /* List of material models */
                                           1 @ /* Number of material models */

    IF( status != 0 ) THEN
        write("Call to db_create_valid_const_models failed.")
        msg_to_form( status, 4, appcode(status), 1, 1., "" )
        RETURN
    END IF

    /*
    * Define material categories
    */

    status = db_create_mtl_magics_defn (***** 1 ***** @ /* Analysis code */
                                       ***** 1 ***** @ /* Analysis type */
                                       ***** 1 ***** @ /* Option category */
                                       "Thermal Expansion" ) /* Label for category */

    IF( status != 0 ) THEN
        write("Call to db_create_mtl_magics_defn failed.")
        msg_to_form( status, 4, appcode(status), 1, 1., "" )
        RETURN
    END IF

```

**EXERCISE 7*****Loading Material Property Attributes***

```
status = db_create_mtl_magics_defn ( ***** 1 ***** @ /* Analysis code */
                                ***** 1 ***** @ /* Analysis type */
                                ***** 1 ***** @ /* Option category */
                                "Damping" ) /* Label for category */

IF( status != 0 ) THEN
write("Call to db_create_mtl_magics_defn failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

/*
* define the items(options) which will be placed inside each of the options
*/

status = db_create_mtl_magics_options ( ***** 2 *****@ /* Analysis code */
                                       ***** 2 ***** @ /* Option (Add pg. 7-49) */
                                       ***** 2 ***** ) /* Label */

IF( status != 0 ) THEN
write("Call to db_create_mtl_magics_options failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_mtl_magics_options ( ***** 2 *****@ /* Analysis code */
                                       ***** 2 ***** @ /* Option (None pg. 7-49) */
                                       ***** 2 ***** ) /* Label */

IF( status != 0 ) THEN
write("Call to db_create_mtl_magics_options failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

/*
* Group/place the items inside each of the categories
*/

status = db_create_mtl_allowable_magics ( ***** 3 *****@ /* Analysis code */
                                          ***** 3 ***** @ /* Material Category */
                                          ***** 3 ***** @ /* Option Category ID */
                                          ***** 3 ***** @ /* List of material options */
                                          ***** 3 ***** ) /* Number of material options */

IF( status != 0 ) THEN
write("Call to db_create_mtl_allowable_magics failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_mtl_allowable_magics ( ***** 3 ***** @ /* Analysis code */
                                          ***** 3 ***** @ /* Material Category */
                                          ***** 3 ***** @ /* Option Category ID */
                                          ***** 3 ***** @ /* List of material options */
                                          ***** 3 ***** ) /* Number of material options */

IF( status != 0 ) THEN
write("Call to db_create_mtl_allowable_magics failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

/*
* Tie the categories to the material model (i.e. lines, non-linear ...)
*/
```

```

status = db_create_mtl_magics ( ***** 4 ***** @ /* Analysis code */
                             ***** 4 ***** @ /* Analysis type */
                             ***** 4 ***** @ /* Material Category */
                             ***** 4 ***** @ /* Material Constitutive Model */
                             ***** 4 ***** @ /* List of Option categories */
                             2 ) /* Number of Option categories */

/*
* Define material words
*/

status = db_create_matl_prop_alias ( anal_code_id, @ /* Analysis code */
                                     _STRUCTURAL, @ /* Analysis type */
                                     1, @ /* Material category */
                                     1, @ /* Material Constitutive model */
                                     _REF_TEMP, @ /* Material Word */
                                     "Reference Temperature", @ /* Label (Alias) */
                                     _NO_FIELD ) /* Type of field */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_matl_prop_alias ( anal_code_id, @ /* Analysis code */
                                     _STRUCTURAL, @ /* Analysis type */
                                     1, @ /* Material category */
                                     1, @ /* Material Constitutive model */
                                     _ELASTIC_MODULUS, @ /* Material Word */
                                     "Young's Modulus", @ /* Label (Alias) */
                                     _TEMP_DEP_FIELD ) /* Type of field */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_matl_prop_alias ( anal_code_id, @ /* Analysis code */
                                     _STRUCTURAL, @ /* Analysis type */
                                     1, @ /* Material category */
                                     1, @ /* Material Constitutive model */
                                     _POISSONS_RATIO, @ /* Material Word */
                                     "Poisson's Ratio", @ /* Label (Alias) */
                                     _TEMP_DEP_FIELD ) /* Type of field */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_matl_prop_alias ( anal_code_id, @ /* Analysis code */
                                     _STRUCTURAL, @ /* Analysis type */
                                     1, @ /* Material category */
                                     1, @ /* Material Constitutive model */
                                     _DENSITY, @ /* Material Word */
                                     "Mass Density", @ /* Label (Alias) */
                                     _TEMP_DEP_FIELD ) /* Type of field */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

```

**EXERCISE 7*****Loading Material Property Attributes***

```

status = db_create_matl_prop_alias ( anal_code_id,    @ /* Analysis code          */
                                   _STRUCTURAL,      @ /* Analysis type           */
                                   1,                @ /* Material category        */
                                   1,                @ /* Material Constitutive model */
                                   _THERMAL_EXPANSION, @ /* Material Word           */
                                   "Thermal Expansion Coeff.", @ /* Label (Alias)           */
                                   _TEMP_DEP_FIELD    ) /* Type of field            */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_matl_prop_alias ( anal_code_id,    @ /* Analysis code          */
                                   _STRUCTURAL,      @ /* Analysis type           */
                                   1,                @ /* Material category        */
                                   1,                @ /* Material Constitutive model */
                                   _STRUCTURAL_DAMPING, @ /* Material Word           */
                                   "Structural Damping Coeff.", @ /* Label (Alias)           */
                                   _TEMP_DEP_FIELD    ) /* Type of field            */

IF( status != 0 ) THEN
write("Call to db_create_matl_prop_alias failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

/*
* Define the material records
*/

status = db_create_allowable_matl_prop ( ***** 5 *****@ /* Analysis code          */
                                       ***** 5 ***** @ /* Analysis type           */
                                       ***** 5 ***** @ /* Material Category        */
                                       ***** 5 ***** @ /* Material Constitutive model */
                                       ***** 5 ***** @ /* List of 5 option categories */
                                       @
                                       @
                                       @
                                       @
                                       @
                                       @ /* List of material words   */
                                       ***** 5 ***** ) /* Number of material words */

IF( status != 0 ) THEN
write("Call to db_create_allowable_matl_prop failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_allowable_matl_prop ( ***** 5 *****@ /* Analysis code          */
                                       ***** 5 ***** @ /* Analysis type           */
                                       ***** 5 ***** @ /* Material Category        */
                                       ***** 5 ***** @ /* Material Constitutive model */
                                       ***** 5 ***** @ /* List of 5 option categories */
                                       @
                                       @
                                       @
                                       @
                                       @ /* List of material words   */
                                       ***** 5 ***** ) /* Number of material words */

IF( status != 0 ) THEN
write("Call to db_create_allowable_matl_prop failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

```

```
status = db_create_allowable_matl_prop ( ***** 5 *****@ /* Analysis code */
***** 5 ***** @ /* Analysis type */
***** 5 ***** @ /* Material Category */
***** 5 ***** @ /* Material Constitutive model */
***** 5 ***** @ /* List of 5 option categories */
***** 5 ***** @
***** 5 ***** @
***** 5 ***** @
***** 5 ***** @ /* List of material words */
***** 5 ***** ) /* Number of material words */

IF( status != 0 ) THEN
write("Call to db_create_allowable_matl_prop failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF

status = db_create_allowable_matl_prop ( ***** 5 *****@ /* Analysis code */
***** 5 ***** @ /* Analysis type */
***** 5 ***** @ /* Material Category */
***** 5 ***** @ /* Material Constitutive model */
***** 5 ***** @ /* List of 5 option categories */
***** 5 ***** @
***** 5 ***** @
***** 5 ***** @ /* List of material words */
***** 5 ***** ) /* Number of material words */

IF( status != 0 ) THEN
write("Call to db_create_allowable_matl_prop failed.")
msg_to_form( status, 4, appcode(status), 1, 1., "" )
RETURN
END IF
$ Set the new Miracle to be the Default Analysis Code
uil_pref_analysis.set_analysis_pref( "Miracle", "Structural", ".inp", ".ans" )
$ End
END FUNCTION /* load_miracle */
```

## Solutions

```

1)status = db_create_mtl_magics_defn ( anal_code_id, @ /* Analysis code */
20, @ /* Option category */
/*Thermal Expansion*/ ) /* Label for category */
status = db_create_mtl_magics_defn ( anal_code_id, @ /* Analysis code */
21, @ /* Option category */
/*STRUCTURAL, @ /* Analysis type */
status = db_create_mtl_magics_options ( anal_code_id, @ /* Analysis code */
31, @ /* Option (Add pg. 7-49) */
/*exclude*/ /* Label */
status = db_create_mtl_magics_options ( anal_code_id, @ /* Analysis code */
30, @ /* Option (None pg. 7-49) */
/*include*/ /* Label */
3)status = db_create_mtl_allowable_magics ( anal_code_id, @ /* Analysis code */
20, @ /* Option Category ID */
[30,31], @ /* List of material options */
2 ) /* Number of material options */
status = db_create_mtl_allowable_magics ( anal_code_id, @ /* Analysis code */
MAT_CAT_ISOTROPIC, @ /* Material Category */
21, @ /* Option Category ID */
[30,31], @ /* List of material options */
2 ) /* Number of material options */
4)status = db_create_mtl_magics ( anal_code_id, @ /* Analysis code */
/*STRUCTURAL, @ /* Analysis type */
MAT_CAT_ISOTROPIC, @ /* Material Category */
1, @ /* Material Constitutive Model */
[20, 21 ], @ /* List of Option categories */
2 ) /* Number of Option categories */
5)status = db_create_allowable_mtl_prop ( anal_code_id, @ /* Analysis code */
/*STRUCTURAL, @ /* Analysis type */
1, @ /* Material Category */
1, @ /* Material Constitutive model */
[30,30,0,0], @ /* List of 5 option categories */
[REF_TEMP, @
ELASTIC MODULUS, @
POISSONS RATIO, @
DENSITY, @
THERMAL_EXPANSION, @
/*STRUCTURAL DAMPING], @ /* List of material words */
6 ) /* Number of material words */

```

```

status = db.create_allowable_matl_prop ( anal_code_id, @ // * Analysis code *
STRUC_TURAT, @ // * Analysis type *
1, @ // * Material Category *
1, @ // * Material Constitutive model *
[31,31,0,0,0], @ // List of 5 option categories *
@
ELASTIC_MODULUS, @
POISSONS_RATIO, @
DENSITY, @
STRUC_TURAT_DAMPING], @ // List of material words *
4 ) // * Number of material words *
status = db.create_allowable_matl_prop ( anal_code_id, @ // * Analysis code *
STRUC_TURAT, @ // * Analysis type *
1, @ // * Material Category *
1, @ // * Material Constitutive model *
[31,30,0,0,0], @ // List of 5 option categories *
@
ELASTIC_MODULUS, @
POISSONS_RATIO, @
DENSITY, @
THERMAL_EXPANSION ], @ // List of material words *
5 ) // * Number of material words *
status = db.create_allowable_matl_prop ( anal_code_id, @ // * Analysis code *
STRUC_TURAT, @ // * Analysis type *
1, @ // * Material Category *
1, @ // * Material Constitutive model *
[30,31,0,0,0], @ // List of 5 option categories *
@
REF_TEMP, @
ELASTIC_MODULUS, @
POISSONS_RATIO, @
DENSITY, @
THERMAL_EXPANSION ], @ // List of material words *
5 ) // * Number of material words *

```