

## Project tasks

In this example we will create an application used for managing Project tasks. It's composed of two forms, the first one contains three tabs and from these tabs we can create projects, subprojects and task groups. The second form is composed of two tabs, the first one is for creating tasks and the second one for tasks search within selected criteria.

If we create a project, subproject or a task group using the first form, it will automatically save this item to the database "projecttask.db" and refresh the combo box items in the second form. Second form combo boxes are filled in with data from the database also. Forming of a new task will insert data into the "task" table of database.

```
base := "projecttask.db"
```

Name of the database  
file we are working with

```
w := widget(0, "Create Project")  
set size(w, 650, 200)
```

Creates a widget and  
sets its size

```
projname := database query(base, "SELECT Name FROM project ORDER BY Name ASC")
```

Creates a query string that will be executed in  
the function, **database query**

We want to determine the names of projects  
from a database

```
w1 := widget(0, "Project")
```

```
set size(w1, 600, 100)
```

```
l11 := label(w1, "Project Name")
```

```
set pos(l11, 230, 30)
```

```
li11 := line text box(w1, "")
```

```
set pos(li11, 230, 50)
```

```
b1 := button(w1, "Create Project")
```

```
set pos(b1, 270, 90)
```

```
w2 := widget(0, "Subproject")
```

```
set size(w2, 600, 100)
```

```
l21 := label(w2, "Select Project")
```

```
set pos(l21, 130, 30)
```

```
li21 := combo box(w2, projname)
```

```
set pos(li21, 130, 50)
```

```
l22 := label(w2, "Subproject Name")
```

```
set pos(l22, 300, 30)
```

```
li22 := line text box(w2, "")
```

```
set pos(li22, 300, 50)
```

```
b2 := button(w2, "Create Subproject")
```

```
set pos(b2, 270, 90)
```

Creates the widget for  
the project and  
subproject tabs

```

w3 := widget(0 , "Tast Group")
set pos(w3 , 600 , 100)

l31 := label(w3 , "Select Project")
set pos(l31 , 100 , 30)

li31 := combo box(w3 , projname)
set pos(li31 , 100 , 50)

```

Creates the widget for the task group tab

```

qq1 := "SELECT subproject.Name FROM subproject INNER JOIN project ON project.ID = "
qq1 += "subproject.proj_id WHERE project.Name = " + widget value(l31) + ""
subprojname := database query(base , qq1)

```

```

on event(li31 , reload())

```

Creates a query that returns the subprojects names from a selected project

```

l32 := label(w3 , "Select Subproject")
set pos(l32 , 230 , 30)

li32 := combo box(w3 , subprojname)
set pos(li32 , 230 , 50)

l33 := label(w3 , "Task Group Name")
set pos(l33 , 350 , 30)

li33 := line text box(w3 , "")
set pos(li33 , 350 , 50)

b3 := button(w3 , "Create Task Group")
set pos(b3 , 270 , 90)

```

On the change of li31 choice, run the function, reload()

```

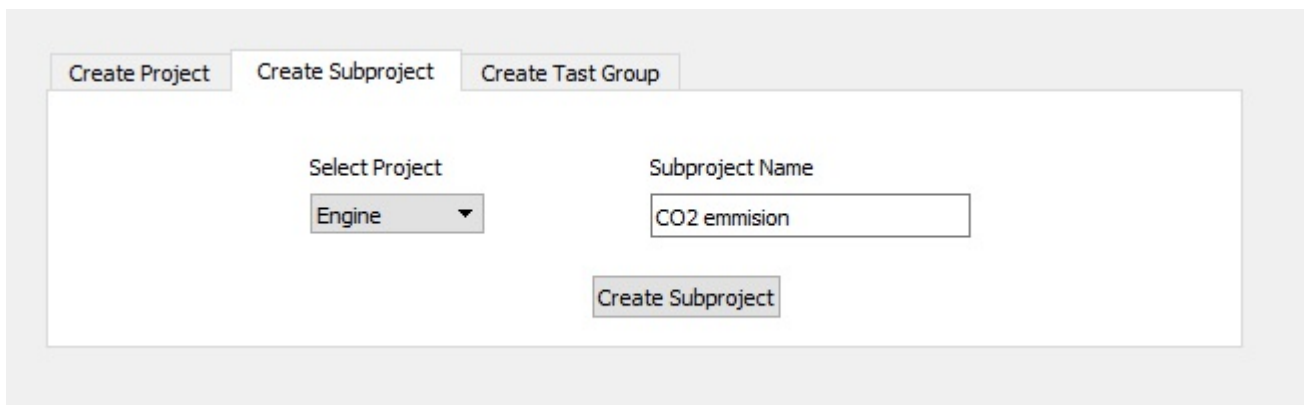
reload( )
{
1 qq2 := "SELECT subproject.Name FROM subproject INNER JOIN project ON project.ID = "
2 qq2 += "subproject.proj_id WHERE project.Name = " + widget value(l31) + ""
3 subproj := database query(base , qq2)
  if(is undefined(subproj))
4 {
5   1 set widget value(li32 , "")
6 }
  else
7 {
8   1 set widget value(li32 , subproj)
9 }
}

```

```

tab11 := tab widget( w , [ "Create Project" , "Create Subproject" , "Create Tast Group" ] , [ w1 , w2 , w3 ] )
set pos(tab11 , 20 , 20)
set size(tab11 , 600 , 150)

```



```
ww := widget(0 , "Tasks")
```

```
set size(ww , 650 , 550)
```

```
w4 := widget(0 , "Create Task")
```

```
set size(w4 , 600 , 450)
```

```
l41 := label(w4 , "Select Project")
```

```
set pos(l41 , 70 , 30)
```

```
li41 := combo box(w4 , projname)
```

```
set pos(li41 , 70 , 50)
```

Creates the widget for tasks tabs

```
qq3 := "SELECT subproject.Name FROM subproject INNER JOIN project ON project.ID = "
```

```
qq3 += "subproject.proj_id WHERE project.Name = " + widget value(li41) + ""
```

```
subprojn := database query(base , qq3)
```

```
on event(li41 , reload1( ))
```

```
l42 := label(w4 , "Select Subproject")
```

```
set pos(l42 , 240 , 30)
```

```
li42 := combo box(w4 , subprojn)
```

```
set pos(li42 , 240 , 50)
```

```
l43 := label(w4 , "Select Task Group")
```

```
set pos(l43 , 400 , 30)
```

Creates a query that returns subprojects names from the selected project

Creates query that returns the task project names from the selected subproject

```
qq6 := "SELECT dgroup.Name FROM dgroup INNER JOIN subproject ON subproject.ID = "
```

```
qq6 += "dgroup.subproj_id WHERE subproject.Name = " + widget value(li42) + ""
```

```
groupn := database query(base , qq6)
```

```
on event(li42 , reload2( ))
```

```
li43 := combo box(w4 , groupnn)
set pos(li43 , 400 , 50)
set size(li43 , 120 , 20)
```

```
g1 := group box(w4 , "Task")
set pos(g1 , 50 , 100)
set size(g1 , 500 , 300)
l44 := label(w4 , "Name")
set pos(l44 , 80 , 130)
li44 := line text box(w4 , "")
set pos(li44 , 80 , 150)
l45 := label(w4 , "Team")
set pos(l45 , 280 , 130)
```

Creates a tasks group box

```
qq8 := "SELECT Name FROM team ORDER BY Name ASC"
```

```
teamn := database query(base , qq8)
```

```
li45 := combo box(w4 , teamn)
```

```
set pos(li45 , 280 , 150)
```

```
l46 := label(w4 , "Person")
```

```
set pos(l46 , 380 , 130)
```

```
qq9 := "SELECT Name FROM person ORDER BY Name ASC"
```

```
personn := database query(base , qq9)
```

```
li46 := combo box(w4 , personn)
```

```
set pos(li46 , 380 , 150)
```

```
l47 := label(w4 , "Start Date (YYYY-MM-DD)")
```

```
set pos(l47 , 100 , 200)
```

```
li47 := line text box(w4 , "")
```

```
set pos(li47 , 100 , 220)
```

```
l48 := label(w4 , "End Date (YYYY-MM-DD)")
```

```
set pos(l48 , 340 , 200)
```

```
li48 := line text box(w4 , "")
```

```
set pos(li48 , 340 , 220)
```

Creates a query that returns the team names from the database for a list box populating

Creates a query that returns the person's names from the database for a list box populating

Creates text boxes used for dates

```

l49 := label(w4 , "Completed")
set pos(l49 , 270 , 260)
li49 := double spin box(w4 , 0.0 , 100.0 , 0.1 , 0.0)
set pos(li49 , 270 , 280)
l410 := label(w4 , "Note")
set pos(l410 , 200 , 320)
li410 := text box(w4 , "")
set pos(li410 , 200 , 340)
set size(li410 , 210 , 50)
b4 := button(w4 , "Create Task")
set pos(b4 , 270 , 420)
on event(b4 , taskCr())
w5 := widget(0 , "Search Task")
set size(w5 , 600 , 450)
l51 := label(w5 , "Select Criteria")
set pos(l51 , 270 , 50)

```

Creates a spin box and a text box for form

Creates a button and calls the function ,taskCr(), on submit

Creates a widget for the search tab and customizes it

```

qlist := [
    "All tasks"
    "Active tasks list"
    "Tasks finished 50 percent or less"
]

```

```

li51 := combo box(w5 , qlist)
set pos(li51 , 220 , 70)
set widget value(li51 , "")

```

```

stv := [ "" "" "" "" "" "" "" "" ]

```

```

tt := table create(stv , colT)

```

```

tab := table widget(w5 , tt)
set pos(tab , 0 , 110)
set size(tab , 600 , 350)

```

```

col := [
    "Name"
    "Team"
    "Person"
    "Start date"
    "End date"
    "Completed"
    "Note"
    ⋮
]

```

```

tab2 := tab widget( ww , [ "Create Task" ] , [ w4 ] )
                    [ "Search Tasks" ] , [ w5 ] )
set pos(tab2 , 20 , 20)
set size(tab2 , 600 , 500)

```

```

reload1( )

```

```

{
1 qq4 := "SELECT subproject.Name FROM subproject INNER JOIN project ON project.ID = "
2 qq4 += "subproject.proj_id WHERE project.Name = " + widget value(li41) + ""
3 subproj := database query(base , qq4)
  if(is undefined(subproj))
4 {
  1 set widget value(li42 , "")
  }
  else
5 {
  1 set widget value(li42 , subproj)
  }
6 qq5 := "SELECT dgroup.Name FROM dgroup INNER JOIN subproject ON subproject.ID = "
7 qq5 += "dgroup.subproj_id WHERE subproject.Name = " + widget value(li42) + ""
8 tgroup := database query(base , qq5)
  if(is undefined(tgroup))
9 {
  1 set widget value(li43 , "")
  }
  else
10 {
  1 set widget value(li43 , tgroup)
  }
11
}

```

Function ,reload1(), is used to populate subprojects and tasks groups lists based on the selected option from the project list

```

reload2( )
{
1 qq7 := "SELECT dgroup.Name FROM dgroup INNER JOIN subproject ON subproject.ID = "
2 qq7 += "dgroup.subproj_id WHERE subproject.Name = " + widget value(li42) + ""
3 tgroup := database query(base , qq7)
  if(is undefined(tgroup))
4 {
  1 set widget value(li43 , "")
  }
  else
5 {
  1 set widget value(li43 , tgroup)
  }
6
}

```

Function ,reload2(), is used to populate tasks groups lists based on the selected option from the subproject list

## Main form

It has two tabs, one to create tasks from it and the other to search created tasks

Create Task Search Tasks

Select Project  
Engine

Select Subproject  
Emission

Select Task Group  
Exhaust gasses

Task

Name  
CO2 emmision

Team  
Team 2

Person  
David Man

Start Date (YYYY-MM-DD)  
2018-05-12

End Date (YYYY-MM-DD)  
2019-07-01

Complited  
25.5

Note

Create Task



```
taskCr( )
```

```
{  
1 tm := "SELECT team.ID FROM team WHERE team.Name = " + widget value(li45) + ""  
2 tm1 := database query(base , tm)  
3 per := "SELECT person.ID FROM person WHERE person.Name = " + widget value(li46) + ""  
4 per1 := database query(base , per)  
5 gr := "SELECT dgroup.ID FROM dgroup WHERE dgroup.Name = " + widget value(li43) + ""  
6 gr1 := database query(base , gr)  
7 qq10 := "INSERT INTO 'task' (Name, team_id, person_id, start_date, end_date, completed"  
8 qq10 += ", note, group_id) VALUES (" + widget value(li44) + ", " + tm1 + ", " + per + ", "  
9 qq10 += widget value(li47) + ", " + widget value(li48) + ", " + widget value(li49) + ", "  
10 qq10 += widget value(li410) + ", " + gr1 + ")"  
11 database query(base , qq10)  
12 set tooltip(li44 , widget value(li44))  
13 set tooltip(li47 , widget value(li47))  
14 set tooltip(li48 , widget value(li48))  
15 set tooltip(li49 , widget value(li49))  
16 set tooltip(li410 , widget value(li410))  
17 set widget value(li44 , "")  
18 set widget value(li47 , "")  
19 set widget value(li48 , "")  
20 set widget value(li49 , 0.0)  
21 set widget value(li410 , "")  
}
```

Function ,reload2(), is used to populate tasks groups lists based on the selected option from the subproject list

```
on event(b1 , insertProj())
```

```
insertProj( )
```

```
{  
1 qq20 := "SELECT project.Name FROM project WHERE project.Name = " + widget value(li11)  
2 qq20 += ""  
3 prov1 := database query(base , qq20)  
4 qq21 := "INSERT INTO project (Name) VALUES (" + widget value(li11) + """)"  
  if(is undefined(prov1))  
  {  
    1 database query(base , qq21)  
    2 qqq := "SELECT Name FROM project ORDER BY Name ASC"  
    3 projn := database query(base , qqq)  
    4 set widget value(li21 , projn)  
    5 set widget value(li31 , projn)  
    6 set widget value(li41 , projn)  
    7 set tooltip(li11 , widget value(li11))  
    8 set widget value(li11 , "")  
  }  
  else  
  {  
    1 message box("Create Project" , "Project with the same name already exists!")  
  }  
}
```

Function insertProj() is used to insert a project in a database by the pressing the button

```
on event(b2 , insertSubproj(0) )
```

```
insertSubproj( )
```

```
{  
  1 qq30 := "SELECT subproject.Name FROM subproject WHERE subproject.Name = "  
  2 qq30 += widget value(li22) + ""  
  3 prov2 := database query(base , qq30)  
  4 qq31 := "SELECT project.ID FROM project WHERE project.Name = "  
  5 qq31 += widget value(li21) + ""  
  6 projID := database query(base , qq31)  
  7 qq32 := "INSERT INTO subproject (Name, proj_id) VALUES (" + widget value(li22) + ", "  
  8 qq32 += projID + ""  
  if(is undefined(prov2))  
  {  
    1 database query(base , qq32)  
    2 qqq := "SELECT Name FROM project ORDER BY Name ASC"  
    3 projn := database query(base , qqq)  
    4 set widget value(li31 , projn)  
    5 set widget value(li41 , projn)  
    6 set tooltip(li22 , widget value(li22))  
    7 set widget value(li22 , ""  
  }  
  else  
  {  
    10 1 message box("Create Subproject" , "Subproject with the same name already exists!")  
  }  
}
```

Function insertSubproj() is used to insert a subproject in a database by pressing the button

```

on event(b3 , insertGroup(0))
insertGroup( )
{
1 qq40 := "SELECT dgroup.Name FROM dgroup WHERE dgroup.Name = " + widget value(li33)
2 qq40 += ""
3 prov3 := database query(base , qq40)
4 qq41 := "SELECT ID FROM subproject WHERE Name = " + widget value(li32) + ""
5 subprojID := database query(base , qq41)
6 qq42 := "INSERT INTO dgroup (Name, subproj_id) VALUES (" + widget value(li33) + ", "
7 qq42 += subprojID + ")"
  if(is undefined(prov3))
  {
8     1 database query(base , qq42)
     2 qqq := "SELECT Name FROM project ORDER BY Name ASC"
     3 taskn := database query(base , qqq)
     4 set widget value(li41 , taskn)
     5 set tooltip(li33 , widget value(li33))
     6 set widget value(li33 , "")
  }
  else
  {
9     1 message box("Create Task Group" , "Task group with the same name already exists!")
  }
}

```

Function insertGroup() is used to insert a task group into a database by pressing the button

```

firstOp( )
{
1 qqq1 := "SELECT task.Name as 'Task', team.Name as 'Team', person.Name as 'Person', "
2 qqq1 += "task.start_date as 'Start Date', task.end_date as 'End Date', task.completed as "
3 qqq1 += "'Completed', task.note as 'Note', project.Name as 'Project', subproject.Name as "
4 qqq1 += "'Subproject', dgroup.Name as 'Task Group' FROM task INNER JOIN team ON "
5 qqq1 += "team.ID = task.team_id INNER JOIN person ON person.ID = task.person_id "
6 qqq1 += "INNER JOIN dgroup ON dgroup.ID = task.group_id INNER JOIN project ON "
7 qqq1 += "project.ID = dgroup.proj_id INNER JOIN subproject ON subproject.ID = "
8 qqq1 += "dgroup.subproj_id"
9 val1 := database query(base , qqq1)
10 return(val1)
}

```

secondOp( )

```
{  
1  qq2 := "SELECT task.Name as 'Tast', team.Name as 'Team', person.Name as 'Person', "  
2  qq2 += "task.start_date as 'Start Date', task.end_date as 'End Date', task.completed as "  
3  qq2 += "'Completed', task.note as 'Note', project.Name as 'Project', subproject.Name as "  
4  qq2 += "'Subproject', dgroup.Name as 'Task Group' FROM task INNER JOIN team ON "  
5  qq2 += "team.ID = task.team_id INNER JOIN person ON person.ID = task.person_id "  
6  qq2 += "INNER JOIN dgroup ON dgroup.ID = task.group_id INNER JOIN project ON "  
7  qq2 += "project.ID = dgroup.proj_id INNER JOIN subproject ON subproject.ID = "  
8  qq2 += "dgroup.subproj_id WHERE date('now') BETWEEN date(task.start_date) AND "  
9  qq2 += "date(task.end_date)"  
10 val2 := database query(base , qq2)  
11 return(val2)  
}
```

thirdOp( )

```
{  
1  qq3 := "SELECT task.Name as 'Tast', team.Name as 'Team', person.Name as 'Person', "  
2  qq3 += "task.start_date as 'Start Date', task.end_date as 'End Date', task.completed as "  
3  qq3 += "'Completed', task.note as 'Note', project.Name as 'Project', subproject.Name as "  
4  qq3 += "'Subproject', dgroup.Name as 'Task Group' FROM task INNER JOIN team ON "  
5  qq3 += "team.ID = task.team_id INNER JOIN person ON person.ID = task.person_id "  
6  qq3 += "INNER JOIN dgroup ON dgroup.ID = task.group_id INNER JOIN project ON "  
7  qq3 += "project.ID = dgroup.proj_id INNER JOIN subproject ON subproject.ID = "  
8  qq3 += "dgroup.subproj_id WHERE task.completed < 50"  
9  val3 := database query(base , qq3)  
10 return(val3)  
}
```

Functions firstOp(), secondOp() and thirdOp() are used to return data from a database by pressing the button

We used them on Search Tasks tab when one of the three predefined criteria is selected

```
on event(li51 , search())
```

```
search( )
```

```
{  
1  if(widget value(li51) == "All tasks")  
   {  
   1  set widget value(tab , firstOp())  
   }  
   else if(widget value(li51) == "Active tasks list")  
   {  
   2  1  set widget value(tab , secondOp())  
   }  
   else if(widget value(li51) == "Tasks finished 50 percent or less")  
   {  
   3  1  set widget value(tab , thirdOp())  
   }  
   else  
   {  
   4  1  set widget value(tab , tt)  
   }  
5 }  
}
```

Function search() calls the functions firstOp(), secondOp() and thirdOp() based on a selected list option

When one of the three predefined criteria is selected these functions return the data from database