

# PROGRAMMING WITH LIVECODE



Higher Computing

USING DATA GRIDS  
HIGHER EXTENSION MATERIAL

# Extension Material: Using Data Grids



A data grid is just like it sounds. It is a way of **displaying** information from **arrays** and **variables** in a structured **grid** which is clear and easy to read.

A data grid allows the user to quickly **sort** data just by clicking on the **column headings**. This function is **automatically built** into the **datagrid** and does not need to be pre-programmed.

You can also use a data grid to display data **without** setting **tabs**. Instead of using a standard output field, arrays are placed into appropriate columns in the datagrid. These are Setup by the **user** in **advance** using the **properties** of the datagrid.

The data grid that you will be using in this extension task holds data from **four arrays** as shown below:

arrayname

arrayram

arrayspeed

arraycost

Graphics Card Name	Memory (Gb) ▼	Speed (Mhz)	Cost (£)
Nvidia 42X	3	1600	575
VaporX	2	870	150
Asus 2	2	790	354
Radeon X2	1	1986	187
GeForce 95	1	550	41
Voodoo 5	1	750	125

You will see that the data grid above has been sorted by **RAM Memory (GB)** in **descending** order. This is done by clicking on the column heading.

## Task 11: ExtremeTech Graphics Card Database

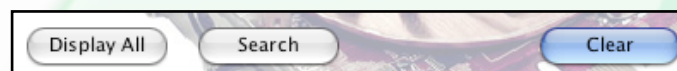
### Specification

A program is required by a company called ExtremeTech. The company specialise in the selling of high quality graphics cards.

ExtremeTech require the program in their retail outlets to allow the customer to **browse** through a list of graphics cards and also allow them to **search** for a graphics card based on their requirements of how much they are willing to spend on a graphics card (**maximum cost**), and how much RAM it must have (**minimum RAM**).

What you have to do:

- Open the stack for “**Using Data Grids**”, the location of this can be found below:
  - [LiveCode Programming Tasks > Extension Task > Using Data Grids.livecode](#)
- **Setup three** buttons on the interface, called “**Display All**”, “**Search**” and “**Clear**”. Double click on each button to give them both a **name**, **label** and **tooltip**.



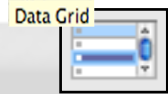
- Open the script for the “**Display All**” and setup **four global arrays** called:
  - **arrayname, arrayram, arrayclockspeed, arraycost**
- Put the following code at the **after** you have setup the arrays within “**Display All**” **button** (internal commentary not required but **do understand** what is going on):

```
// Initialise the variables and fields
set the dgData of group "datagrid1" to empty // Clear the data grid
put empty into field "totalfound" // Clear number of graphics cards found
put empty into field "subheading"
```
- Use the **pseudocode design** on page \* to help you create the rest of the script for this event.

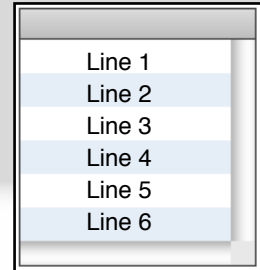
## Task 11: ExtremeTech Graphics Card Database

You are now going to create a **data grid** to go onto the ExtremeTech card. Follow the steps below **carefully**:

**Step 1:** Select the following icon and **drag** it onto the **centre** of the **card**.



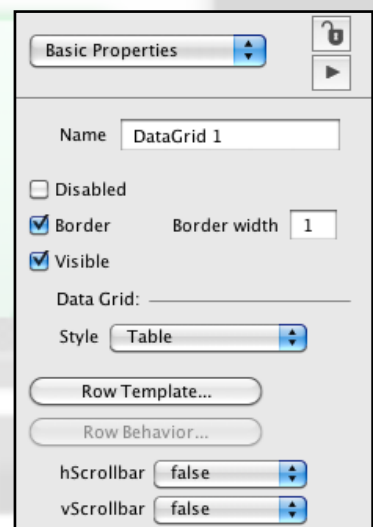
**Step 2:** **Resize** your data grid to make sure it is **6 lines**. This will fit the data you have placed into your arrays on the previous page.



**Step 3:** Double-click on the data grid to bring up its **properties**.

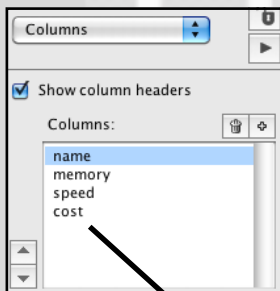
Make sure your settings for the data grid match that of what is shown on the right-hand side.

Ensure that the datagrid name is called **datagrid1** and the **hScrollbar** and **vScrollbar** are set to **false**.

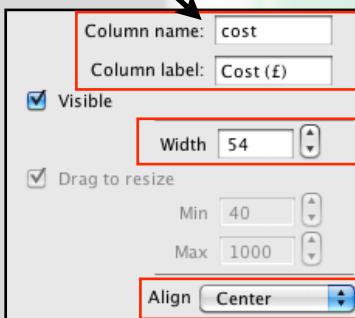


**Step 4:** Select the **Basic Properties** pull down menu and select **Columns**, add **four** columns called:

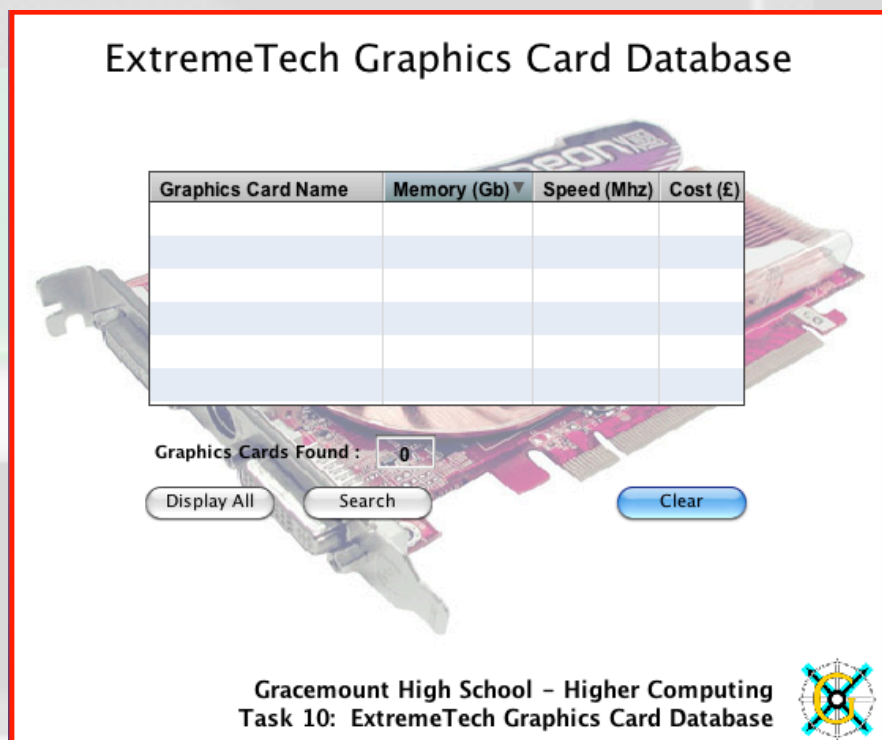
- **name** with a column label of **Graphics Card Name**
- **memory** with a column label of **Memory (GB)**
- **speed** with a column label of **Speed (MHz)**
- **cost** with a column label of **Cost (£)**



Set the **alignment** of all **columns** to **centre** apart from **name** and adjust the **column width** as appropriate.



**Step 5:** Once complete, your **data grid** and **buttons** should look similar to the screenshot shown on the right:



## Task 11: ExtremeTech Graphics Card Database



### Design for “Display All” Button

#### Stepwise Design (the main steps of the program with data flow)

1. Setup
2. Display Cards In: arrayname, arrayram, arrayclockspeed, arraycost

#### Stepwise Refinement (the main steps further refined into smaller steps)

##### 1. Setup

- 1.1 Pass in arrayname, arrayram, arrayclockspeed, arraycost as the global arrays to be used in this event
- 1.2 Setup username as global variable
- 1.4 Clear the “subheading” and “output” fields
- 1.5 Ask the user for their name
- 1.6 Put it into the variable username
- 1.7 Put “RadeonX2”, GeForce95”, “VaporX”, “AsusOX2”, “Nvidia42X” into array\_name
- 1.8 Split arrayname using a comma
- 1.9 Put 1, 1, 2, 2, 3 into arrayram
- 1.10 Split arrayram using a comma
- 1.11 Put 1986, 550, 870, 790, 1600 into arrayclockspeed
- 1.12 Split arrayclockspeed using a comma
- 1.13 Put 187, 41, 150, 354, 575 into arraycost
- 1.14 Split arraycost using a comma

##### 2. Display Cards

- 2.1 Setup the graphicscardinfo and cardcount as local variables
- 2.2 Put empty into the “totalcardsfound” and “subheading” fields
- 2.3 Put the text “Displaying All Graphics Cards” into the “subheading” field
- 2.4 Put 0 into the cardcount field
- 2.5 **Start a Repeat with loop 1 to 5**
- 2.7 Put the arrayname into the name column of graphicscardinfo
- 2.8 Put the arrayram into the memory column of graphicscardinfo
- 2.9 Put the arrayclockspeed into the speed column of graphicscardinfo
- 2.10 Put the arraycost into the cost column of graphicscardinfo
- 2.11 Add 1 to the variable cardcount
- 2.11 **End Repeat**
- 2.12 Copy the contents of group graphicscardinfo into the datagrid named “datagrid1”
- 2.13 Put the cardcount into the field “totalcardsfound”

Please **READ** the following before you begin.

After carefully reading through the design above. You should begin to code the script for the fourth button called “**Display All**”. Key in all of the code **carefully**. The script for this event is on the next page.

Graphics Card Name	Memory (Gb)	Speed (Mhz)	Cost (£)
Nvidia 42X	3	1600	575
VaporX	2	870	150
Asus 2	2	790	354
Radeon X2	1	1986	187
GeForce 95	1	550	41
Voodoo 5	1	750	125

Graphics Cards Found : 6

This event will display all of the data stored in each of the arrays. There are a total of **6 graphics cards** which should be displayed.

## Task 11: ExtremeTech Graphics Card Database

### Implementation: Script "Display All"

Display All

Key in the following code **carefully**. You don't need to include the internal commentary, it is only there to help you understand what is going on.

```
// Allow access to global arrays and variables Setup in main card.
global arrayProductName
global arrayProductRam
global arrayProductClockSpeed
global arrayProductCost
global MaxGraphicsCards

on mouseUp
  display_cards arrayProductName, arrayProductRam, arrayProductClockSpeed,arrayProductCost,
  MaxGraphicsCards // On the same line
end mouseUp

on display_cards arrayProductName, arrayProductRam, arrayProductClockSpeed,
  arrayProductCost, MaxGraphicsCards // On the same line
  // Setup local variables
  local theGraphicsCardInfo
  local CardCount

  // Clear text from the fields
  put empty into field "TotalCardsFound"
  put empty into field "Sub Heading"

  // Display the heading
  put "Displaying All Graphics Cards" into field "Sub Heading"

  // Zero number of graphics cards found
  put 0 into CardCount

  // Display all graphics cards
  repeat with loop = 1 to MaxGraphicsCards

    // Increment the number of graphics cards displayed.
    // This is also used to determine which graphics card details appear within data grid.
    add 1 to CardCount

    // Copy graphics card data from array to data grid
    put arrayProductName[Loop] into theGraphicsCardInfo[CardCount]["name"]
    put arrayProductRam[Loop] into theGraphicsCardInfo[CardCount]["memory"]
    put arrayProductClockSpeed[Loop] into theGraphicsCardInfo[CardCount]["speed"]
    put arrayProductCost[Loop] into theGraphicsCardInfo[CardCount]["cost"]
  end repeat

  set the dgData of group "DataGrid 1" to theGraphicsCardInfo // Copy list to data grid
  put CardCount into Field "TotalCardsFound" // Display number of graphics cards found
end display_cards
```

Now **test** that your script works correctly using the **test data** on the **previous** page.

## Task 11: ExtremeTech Graphics Card Database

### Design for “Search” Button



#### Stepwise Design (the main steps of the program with data flow)

1. Pass in the global arrays and variables
2. Search Cards      **In:** arrayProductName, arrayProductRam, arrayProductClockSpeed, arrayProductCost  
                          **In:** MaxGraphicsCards

#### Stepwise Refinement (the main steps further refined into smaller steps)

1. **Setup**
  - 1.1 Pass in arrayProductName, arrayProductRam, arrayProductClockSpeed, arrayProductCost as the global arrays to be used in this event
  - 1.2 Pass in MaxGraphicsCards as the global variable to be used in this event
2. **Search Cards**
  - 2.1 Setup the theGraphicsCardInfo and CardCount, Min\_RAM and Max\_Cost as local variables
  - 2.2 Put empty into the TotalCardsFound and SubHeading fields
  - 2.3 Put empty into the datagrid
  - 2.4 Put the text “Search Graphics Cards on RAM and Cost” into the SubHeading field
  - 2.5 Put 0 into the CardCount field
  - 2.6 Put empty into theGraphicsCardInfo
  - 2.7 Ask the user for the minimum amount of RAM they require
  - 2.8 Put the users answer into the variable Min\_RAM
  - 2.9 Ask the user for the minimum amount of money they are willing to spend on a graphics card
  - 2.10 Put the users answer into the variable Max\_Cost
  - 2.11 **Start a Repeat with loop 1 to MaxGraphicsCards**
  - 2.12     **If** arrayProductProductRam >= Min\_RAM **and** arrayProductCost <= Max\_Cost **then**
  - 2.13         Add 1 to the CardCount
  - 2.14         Put the arrayProductName into the name column of theGraphicsCardInfo
  - 2.15         Put the arrayProductRam into the memory column of theGraphicsCardInfo
  - 2.16         Put the arrayProductClockSpeed into the speed column of theGraphicsCardInfo
  - 2.17         Put the arrayProductCost into the cost column of theGraphicsCardInfo
  - 2.18     **End If**
  - 2.19 **End Repeat**
  - 2.20 Copy the contents of group theGraphicsCardInfo into the datagrid named “DataGrid 1”
  - 2.21 Put the CardCount into the field “Total Cards Found”

Please **READ** the following before you begin.

After carefully reading through the design above. You should begin to code the script for the “**Search**” button. Key in all of the code **carefully**.

This event will **search** both the **arrayProductRAM** and **arrayProductCost** based on the users **search** of **maximum RAM** and **minimum cost** and display any graphics cards found in the **datagrid**.

After completing the script, you should **test** that the program is working correctly. Enter the **minimum RAM** of “**2**” and a **maximum cost** of “**400**”, the following cards should be displayed in the datagrid:

Graphics Card Name	Memory (Gb) ▾	Speed (Mhz)	Cost (£)
VaporX	2	870	150
Asus 2	2	790	354



## Task 11: ExtremeTech Graphics Card Database

### Implementation: Script "Search"

Key in the following code **carefully**. You don't need to include the internal commentary, it is only there to help you understand what is going on.

```
// Allow access to global arrays and variables Setup in main card.
global arrayname
global arrayram
global arrayclockspeed
global arraycost
global MaxGraphicsCards

on mouseUp
  search_cards
end mouseUp

on search_cards
  // Setup local variables
  local cardcount
  local graphicscardinfo
  local minram
  local maxcost

  // Clear text from the fields
  put empty into field "TotalCardsFound"
  put empty into field "Sub Heading"
  set the dgData of group "DataGrid 1" to empty // Clear the data grid

  // Display the heading
  put "Search Graphics Cards on Minimum RAM and Maximum Cost" into field "Sub Heading"

  put 0 into CardCount // Zero number of graphics cards found
  put empty into theGraphicsCardInfo // Clear graphics card list

  // Ask the user for the minimum cost and maximum amount of RAM required
  ask "Please enter the minimum amount of RAM you wish your graphics card to have:"
  put it into min_ram
  ask "Please enter the maximum amount you are willing to spend on a new graphics card:"
  put it into max_cost

  // Display graphics cards matching search criteria
  repeat with loop = 1 to MaxGraphicsCards
    if arrayProductRam[Loop] >= min_ram AND arrayProductCost[loop] <= max_cost then
      add 1 to CardCount // IF the card matches the search criteria, add card to datagrid
      put arrayProductName[Loop] into theGraphicsCardInfo[CardCount]["name"]
      put arrayProductRam[Loop] into theGraphicsCardInfo[CardCount]["memory"]
      put arrayProductClockSpeed[Loop] into theGraphicsCardInfo[CardCount]["speed"]
      put arrayProductCost[Loop] into theGraphicsCardInfo[CardCount]["cost"]
    end if
  end repeat

  set the dgData of group "DataGrid 1" to theGraphicsCardInfo // Copy list to the data grid
  put CardCount into Field "TotalCardsFound" // Display number of graphics cards found
end search_cards
```

Now **test** that your script works correctly using the **test data** on the **previous** page.