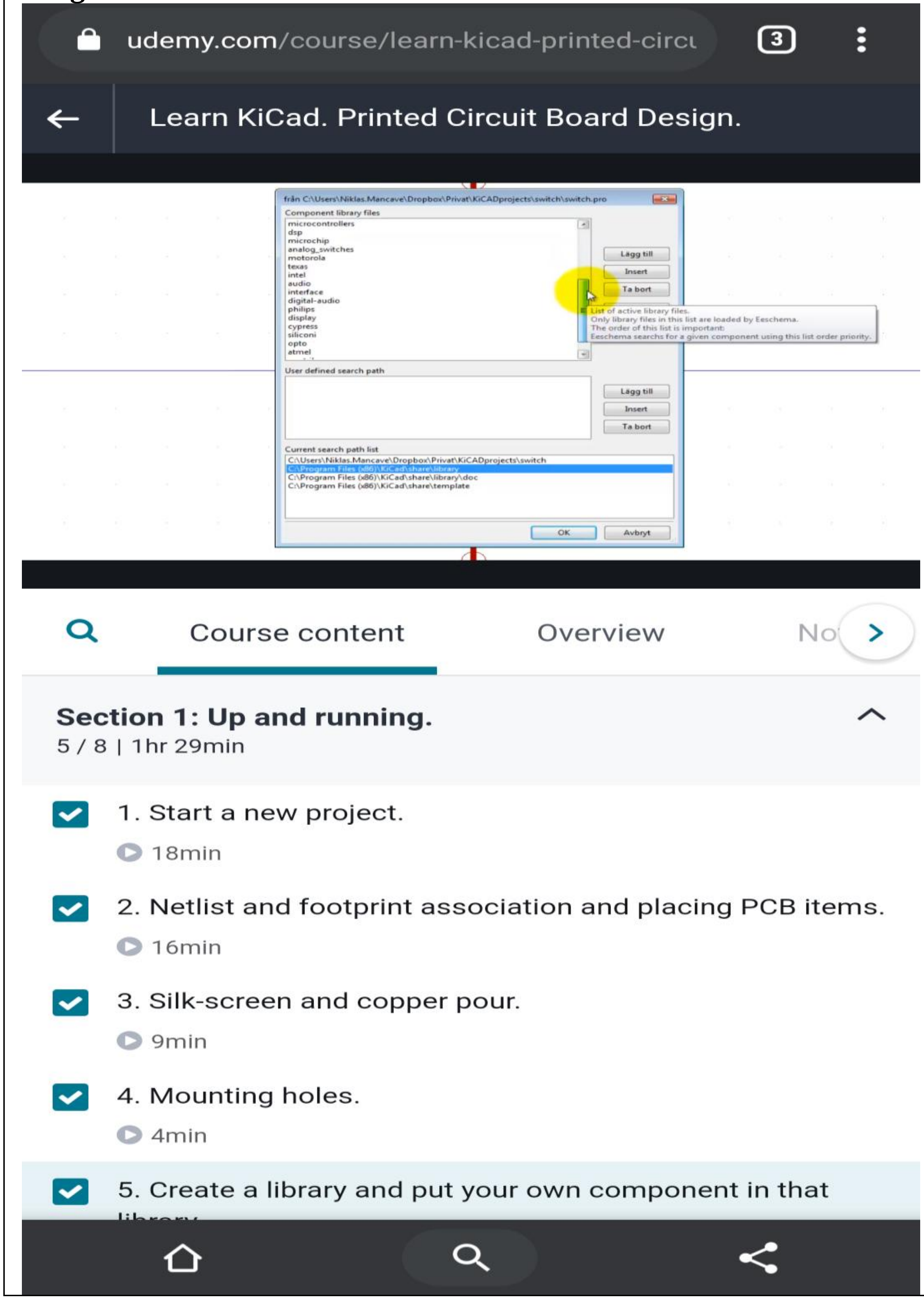
DAILY ASSESSMENT FORMAT

Date:	**12th June 2020**	**Name:**	**Poojary Sushant**
Course:	**Kicad on Udemy**	**USN:**	**4AL18EC400**
Topic:	**A hands –on tour of kicad with a simple project-layout**	**Semester & Section:**	**6th sem 'B"**
Github Repository:	**Sushant7026**		

FORENOON SESSION DETAILS
Image of session



udemy.com/course/learn-kicad-printed-circu

Learn KiCad. Printed Circuit Board Design.

från C:\Users\Niklas.Mancave\Dropbox\Privat\KiCADprojects\switch\switch.pro

Component library files

microcontrollers
dsp
microchip
analog_switches
motorola
texas
intel
audio
interface
digital-audio
philips
display
cypress
siliconi
opto
atmel

List of active library files.
Only library files in this list are loaded by Eeschema.
The order of this list is important:
Eeschema searchs for a given component using this list order priority.

User defined search path

Current search path list

C:\Users\Niklas.Mancave\Dropbox\Privat\KiCADprojects\switch
C:\Program Files (x86)\KiCad\share\library
C:\Program Files (x86)\KiCad\share\library\doc
C:\Program Files (x86)\KiCad\share\template

Section 1: Up and running.
5 / 8 | 1hr 29min

☑ 1. Start a new project.
18min

☑ 2. Netlist and footprint association and placing PCB items.
16min

☑ 3. Silk-screen and copper pour.
9min

☑ 4. Mounting holes.
4min

☑ 5. Create a library and put your own component in that library.

Report –

An optimised PCB Design Flow allows you to create PCB’s that are “Right First Time for Manufacture”. Our customers report that they save time and money by using the Eurocircuits’ Smart Tools. These allow the designer to analyse their design parameters before starting their board layout. Euro circuits developed their Smart Tools to help Electronic Design Engineers achieve an efficient PCB Design Flow. Our Smart Tools provide instant technical validation of design parameters and help guide you towards lower cost options. This results is a PCB layout that is manufacturable at the lowest cost with the highest reliability. Our eC Smart Tools offer a visualisation of the PCB layout which provides a further level of confidence for the designer.The data is checked and any issues are reported (missing files, parameter, data errors etc.).

This allows the designer to view and correct them before the PCB’s are ordered. Our eC Smart Tools also offer you solutions to solve design and data issues quickly and efficiently. This then ensures that the data used for the manufacture of the PCB is Right First Time.Upload your BOM & CPL to the PCBA Visualizer, here you are able to visualise the components placement on the PCB. Using the tools you are able to correct any placement issues to ensure the data used to assemble the PC’s is Right First Time using Eurocircuits Assembly services. Using Eurocircuits’ Smart Tools provides the parameters needed to create a PCB layout that is optimised for manufacture.

Tools menu and Update Schematic :

Allows:

- Display load netlist dialog
- Update PCB from schematic
- Update Footprints from library
- FreeRoute collaboration
- Python scripting console
- External plugins

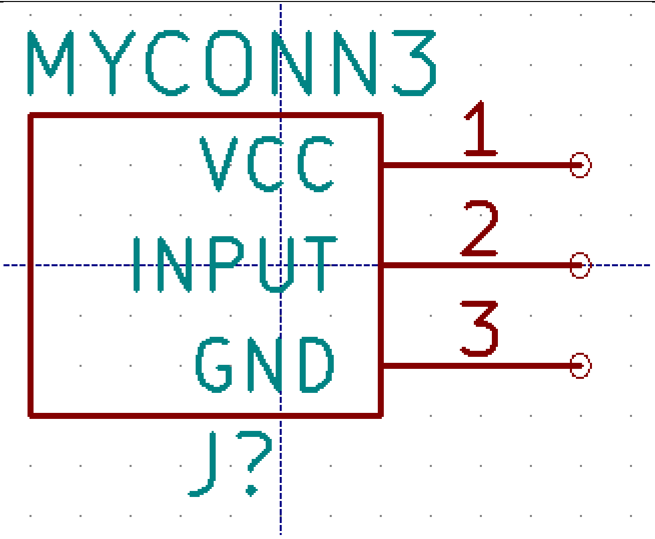
Allows:

- Selection of the footprint libraries.
- Management of general options (units, etc.).
- The management of other display options.
- Creation, editing (and re-read) of the hot keys file.

Sometimes a symbol that you want to place on your schematic is not in a KiCad library. This is quite normal and there is no reason to worry. In this section we will see how a new schematic symbol can be quickly created with KiCad. Nevertheless, remember that you can always find KiCad components on the Internet.

In KiCad, a symbol is a piece of text that starts with 'DEF' and ends with 'ENDDEF'. One or more symbols are normally placed in a library file with the extension *.lib*. If you want to add symbols to a library file you can just use the cut and paste commands of a text editor.

1. We can use the *Component Library Editor* (part of *Eeschema*) to make new components. In our project folder 'tutorial1' let's create a folder named 'library'. Inside we will put our new library file *myLib.lib* as soon as we have created our new component.
2. Now we can start creating our new component. From KiCad, start *Eeschema*, click on the 'Library Editor' icon and then click on the 'New component' icon . The Component Properties window will appear. Name the new component 'MYCONN3', set the 'Default reference designator' as 'J', and the 'Number of units per package' as '1'. Click OK. If the warning appears just click yes. At this point the component is only made of its labels. Let's add some pins. Click on the 'Add Pins' icon on the right toolbar. To place the pin, left click in the centre of the part editor sheet just below the 'MYCONN3' label.
3. In the Pin Properties window that appears, set the pin name to 'VCC', set the pin number to '1', and the 'Electrical type' to 'Power input' then click OK.
4. Place the pin by clicking on the location you would like it to go, right below the 'MYCONN3' label.
5. Repeat the place-pin steps, this time 'Pin name' should be 'INPUT', 'Pin number' should be '2', and 'Electrical Type' should be 'Passive'.
6. Repeat the place-pin steps, this time 'Pin name' should be 'GND', 'Pin number' should be '3', and 'Electrical Type' should be 'Passive'. Arrange the pins one on top of the other. The component label 'MYCONN3' should be in the centre of the page (where the blue lines cross).
7. Next, draw the contour of the component. Click on the 'Add rectangle' icon . We want to draw a rectangle next to the pins, as shown below. To do this, click where you want the top left corner of the rectangle to be (do not hold the mouse button down). Click again where you want the bottom right corner of the rectangle to be.

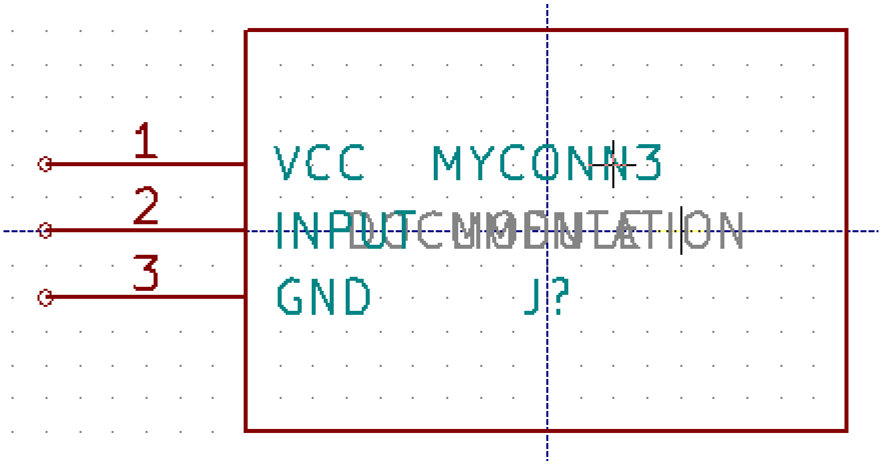


8. If you want to fill the rectangle with yellow, set the fill colour to 'yellow 4' in Preferences → Select color scheme, then select the rectangle in the editing screen with [e], selecting 'Fill background'.
9. Save the component in your library *myLib.lib*. Click on the 'New Library' icon , navigate into *tutorial1/library/* folder and save the new library file with the name *myLib.lib*.
10. Go to Preferences → Component Libraries and add both *tutorial1/library/* in 'User defined search path' and *myLib.lib in* 'Component library files'.
11. Click on the 'Select working library' icon . In the Select Library window click on *myLib* and click OK. Notice how the heading of the window indicates the library currently in use, which now should be *myLib*.
12. Click on the 'Update current component in current library' icon in the top toolbar. Save all changes by clicking on the 'Save current loaded library on disk' icon in the top toolbar. Click 'Yes' in any confirmation messages that appear. The new schematic component is now done and available in the library indicated in the window title bar.
13. You can now close the Component library editor window. You will return to the schematic editor window. Your new component will now be available to you from the library *myLib*.
14. You can make any library *file.lib* file available to you by adding it to the library path. From *Eeschema*, go to Preferences → Library and add both the path to it in 'User defined search path' and *file.lib* in 'Component library files'.

Make schematic components with quicklib

1. Head to the *quicklib* web page:http://kicad.rohrbacher.net/quicklib.php
2. Fill out the page with the following information: Component name: MYCONN3 Reference Prefix: J Pin Layout Style: SIL Pin Count, N: 3

3. Click on the 'Assign Pins' icon. Fill out the page with the following information: Pin 1: VCC Pin 2: input Pin 3: GND. Type : Passive for all 3 pins.
4. Click on the icon 'Preview it' and, if you are satisfied, click on the 'Build Library Component'. Download the file and rename it *tutorial1/library/myQuickLib.lib.*. You are done!
5. Have a look at it using KiCad. From the KiCad project manager, start *Eeschema*, click on the 'Library Editor' icon , click on the 'Import Component' icon , navigate to *tutorial1/library/* and select *myQuickLib.lib.*



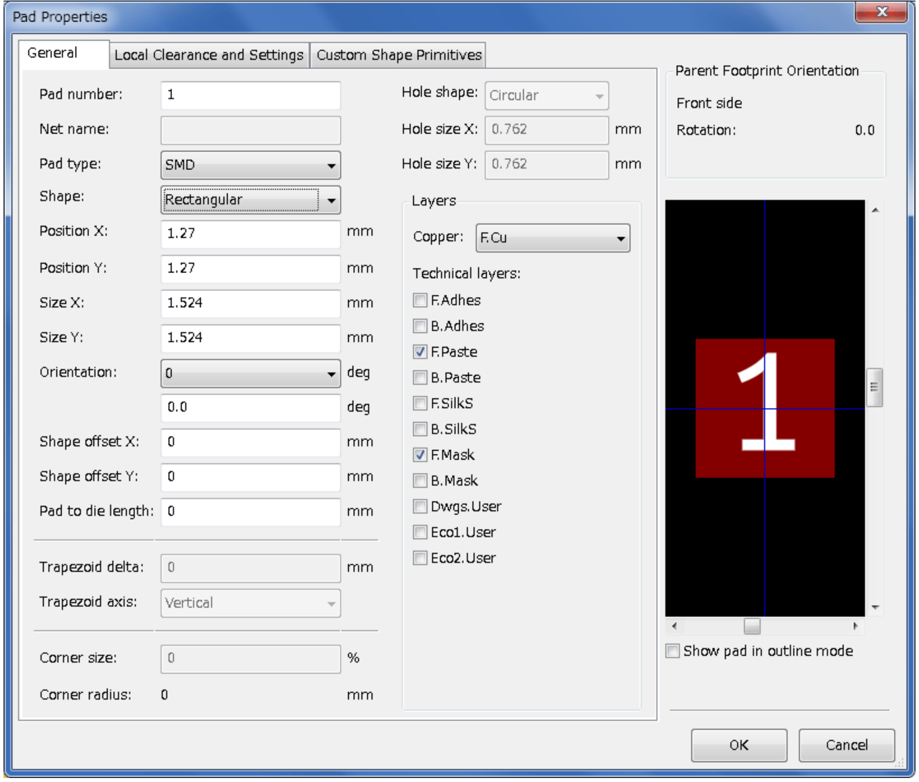
6. You can make this component and the whole library *myQuickLib.lib* available to you by adding it to the KiCad library path. From *Eeschema*, go to Preferences → Component Libraries and add *library* in 'User defined search path' and *myQuickLib.lib* in 'Component library files'.

Using Footprint Editor

1. From the KiCad project manager start the *Pcbnew* tool. Click on the 'Open Footprint Editor' icon on the top toolbar. This will open the 'Footprint Editor'.
2. We are going to save the new footprint 'MYCONN3' in the new footprint library 'myfootprint'. Create a new folder *myfootprint.pretty* in the *tutorial1/* project folder. Click on the Preferences → Footprint Libraries Manager and press 'Append Library' button. In the table, enter "myfootprint" as Nickname, enter "${KIPRJMOD}/myfootprint.pretty" as Library Path and enter "KiCad" as Plugin Type. Press OK to close the PCB Library Tables window. Click on the 'Select active library' icon on the top toolbar. Select the 'myfootprint' library.
3. Click on the 'New Footprint' icon on the top toolbar. Type 'MYCONN3' as the 'footprint name'. In the middle of the screen the 'MYCONN3' label will appear. Under the label you can see the 'REF*' *label. Right click on 'MYCONN3' and move it above 'REF*'*. Right click on 'REF__*', select 'Edit Text' and rename it to 'SMD'. Set the 'Display' value

to 'Invisible'.

4. Select the 'Add Pads' icon on the right toolbar. Click on the working sheet to place the pad. Right click on the new pad and click 'Edit Pad'. You can also use [e].



Pad number: 1
Net name:
Pad type: SMD
Shape: Rectangular
Position X: 1.27 mm
Position Y: 1.27 mm
Size X: 1.524 mm
Size Y: 1.524 mm
Orientation: 0 deg
0.0 deg
Shape offset X: 0 mm
Shape offset Y: 0 mm
Pad to die length: 0 mm
Trapezoid delta: 0 mm
Trapezoid axis: Vertical
Corner size: 0 %
Corner radius: 0 mm

Hole shape: Circular
Hole size X: 0.762 mm
Hole size Y: 0.762 mm

Layers
Copper: F.Cu
Technical layers:
□ F.Adhes
□ B.Adhes
☑ F.Paste
□ B.Paste
□ F.SilkS
□ B.SilkS
☑ F.Mask
□ B.Mask
□ Dwgs.User
□ Eco1.User
□ Eco2.User

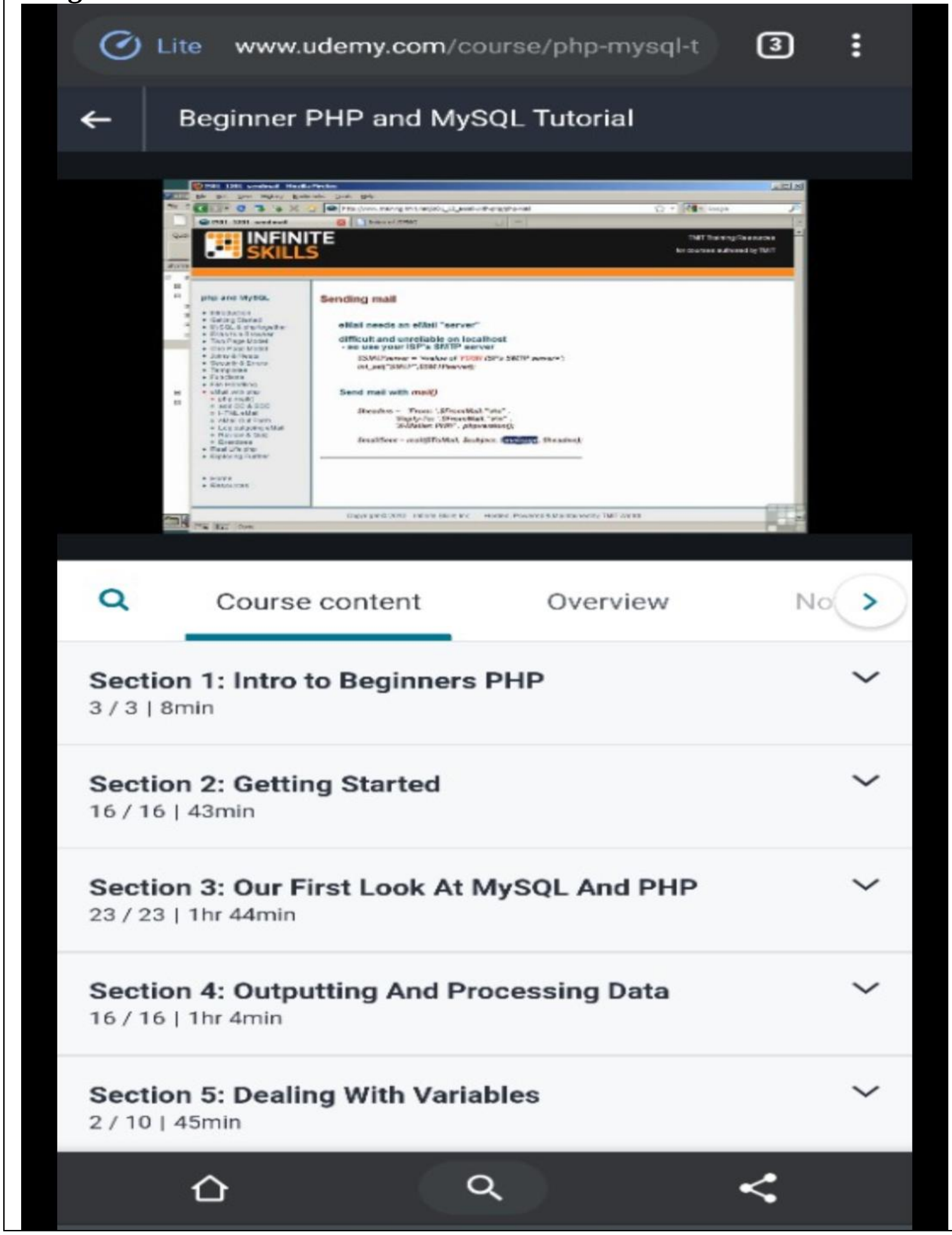
Parent Footprint Orientation
Front side
Rotation: 0.0

□ Show pad in outline mode

5. et the 'Pad Num' to '1', 'Pad Shape' to 'Rect', 'Pad Type' to 'SMD', 'Shape Size X' to '0.4', and 'Shape Size Y' to '0.8'. Click OK. Click on 'Add Pads' again and place two more pads.
6. If you want to change the grid size, Right click → Grid Select. Be sure to select the appropriate grid size before laying down the components.
7. Move the 'MYCONN3' label and the 'SMD' label out of the way so that it looks like the image shown above.
8. When placing pads it is often necessary to measure relative distances. Place the cursor where you want the relative coordinate point *(0,0)* to be and press the space bar. While moving the cursor around, you will see a relative indication of the position of the cursor at the bottom of the page. Press the space bar at any time to set the new origin.
9. Now add a footprint contour. Click on the 'Add graphic line or polygon' button in the right toolbar. Draw an outline of the connector around the component.
10. Click on the 'Save Footprint in Active Library' icon on the top toolbar, using the default name MYCONN3.

Note about portability of KiCad project files What files do you need to send to someone so that they can fully load and use your KiCad project? When you have a KiCad project to share with somebody, it is important that the schematic file *.sch*, the board file *.kicad_pcb*, the project file *.pro* and the netlist file *.net*, are sent together with both the schematic parts file *.lib* and the footprints file *.kicad_mod*. Only this way will people have total freedom to modify the schematic and the board

Date:	**12thJune 2020**	**Name:**	**Poojary Sushant**
Course:	**PHP & MYSQL On Udemy**	**USN:**	**4AL18EC400**
Topic:	**Email with PHP,Real life PHP Introduction**	**Semester & Section:**	**6th & ‘B,**
GitHub Repository :	**Sushant7026**		

AFTERNOON SESSION DETAILS
Image of session 

Lite www.udemy.com/course/php-mysql-t

Beginner PHP and MySQL Tutorial

INFINITE SKILLS

Sending mail

eMail needs an eMail "server"

difficult and unreliable on localhost
- so use your ISP's SMTP server

Send mail with mail()

Course content   Overview   No

Section 1: Intro to Beginners PHP
3 / 3 | 8min

Section 2: Getting Started
16 / 16 | 43min

Section 3: Our First Look At MySQL And PHP
23 / 23 | 1hr 44min

Section 4: Outputting And Processing Data
16 / 16 | 1hr 4min

Section 5: Dealing With Variables
2 / 10 | 45min

Report –

```
<?php
$to = "somebody@example.com";
$subject = "My subject";
$txt = "Hello world!";
$headers = "From: webmaster@example.com" . "\r\n" .
"CC: somebodyelse@example.com";

mail($to,$subject,$txt,$headers);
?>
<?php
$to = "somebody@example.com, somebodyelse@example.com";
$subject = "HTML email";

$message = "
<html>
<head>
<title>HTML email</title>
</head>
<body>
<p>This email contains HTML Tags!</p>
<table>
<tr>
<th>Firstname</th>
<th>Lastname</th>
</tr>
<tr>
<td>John</td>
<td>Doe</td>
</tr>
</table>
</body>
</html>
";

// Always set content-type when sending HTML email
$headers = "MIME-Version: 1.0" . "\r\n";
$headers .= "Content-type:text/html;charset=UTF-8" . "\r\n";

// More headers
$headers .= 'From: <webmaster@example.com>' . "\r\n";
$headers .= 'Cc: myboss@example.com' . "\r\n";

mail($to,$subject,$message,$headers);
?>
```

Building PHP Tools - Embedded Tool

```
Name: <input type="text" name="name" value="<?php echo $name;?>">

E-mail: <input type="text" name="email" value="<?php echo $email;?>">
```

```
Website: <input type="text" name="website" value="<?php echo $website;?>">

Comment: <textarea name="comment" rows="5" cols="40"><?php echo
$comment;?></textarea>

Gender:
<input type="radio" name="gender"
<?php if (isset($gender) && $gender=="female") echo "checked";?>
value="female">Female
<input type="radio" name="gender"
<?php if (isset($gender) && $gender=="male") echo "checked";?>
value="male">Male
<input type="radio" name="gender"
<?php if (isset($gender) && $gender=="other") echo "checked";?>
value="other">Other
```

Managing Deletions - Suspension Fields

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
  die("Connection failed: " . $conn->connect_error);
}

// sql to delete a record
$sql = "DELETE FROM MyGuests WHERE id=3";

if ($conn->query($sql) === TRUE) {
  echo "Record deleted successfully";
} else {
  echo "Error deleting record: " . $conn->error;
}

$conn->close();
?>
```

Managing Deletions-Restore Records

```
<?php
public function restore($id)
    {


        $note = Note::onlyTrashed()->find($id);
        if (!is_null($note)) {

            $note->restore();
            $response = $this->successfulMessage(200, 'Successfully restored', true, $note->count(), $note);
        } else {

            return response($response);
        }
        return response($response);
    }
```