Online Learning Application

Objective: It's a known fact that the Average screen sizes of our phones is increasing, thereby encouraging many to read and learn on the move.

Keeping this trend in mind, you have to develop a Online Learning App, which allows a user to

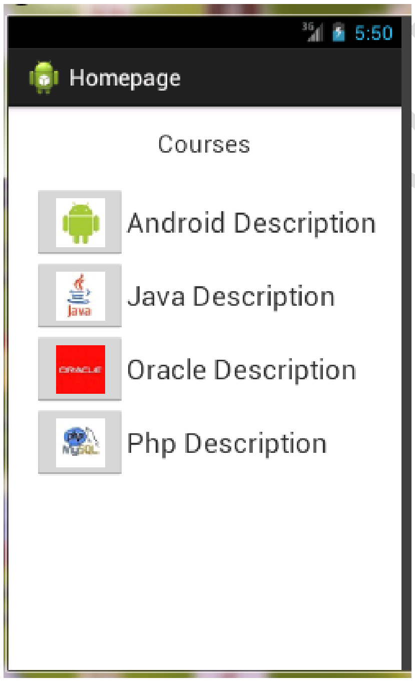
1. Know about all Online learning courses available at your Institution.
2. Add 1 or more courses to the shopping cart.
3. Make Online payment by supplying Bank Name and Bank account details.

Technical Specifications:

1st Activity: Should be a splash screen for 5 seconds which can be created using thread object. After 5 seconds control goes to **2nd Activity.**

2nd Activity : Is the **HomepageActv**. It should provide list of all Courses, as shown in **figure 1 below.**

2nd activity uses <tablelayout>. Each table row contains an imageButton and a Textview. When user clicks on imageButton , transfer control to Another Activity (say AndroidActv.java). This Activity provides description of respective course like this **(refer figure 2)**:



5:50

Homepage

Courses

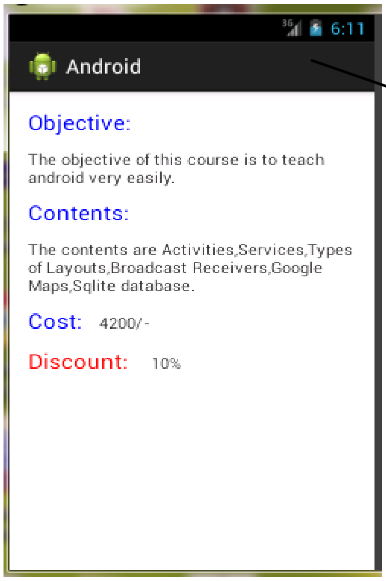
Android Description

Java Description

Oracle Description

Php Description

Figure 1



6:11

Android

Objective:

The objective of this course is to teach android very easily.

Contents:

The contents are Activities,Services,Types of Layouts,Broadcast Receivers,Google Maps,Sqlite database.

Cost: 4200/-

Discount: 10%

Show action Bar

Figure 2

Similarly create 4 such activities (as there are 4 courses on the Home page), let's name them as

- AndroidActv.java (3rd Activity)
- JavaActv.java (4th Activity)
- OracleActv.java (5th Activity)
- PHPActv.java (6th Activity)

In each of above activities

(from 3rd to 6th) code for ActionBar ,

Like this: (refer figure 3)

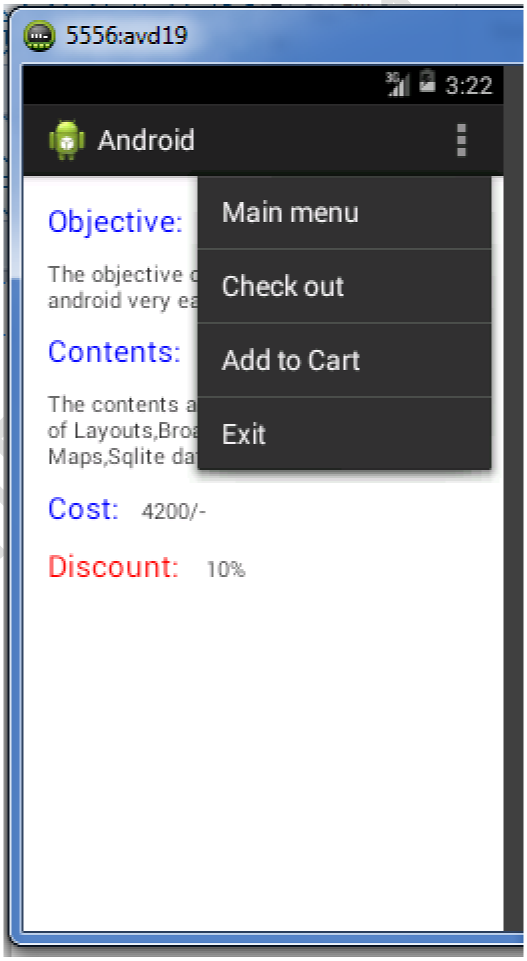


Figure 3

In the options menu there are 4 options:

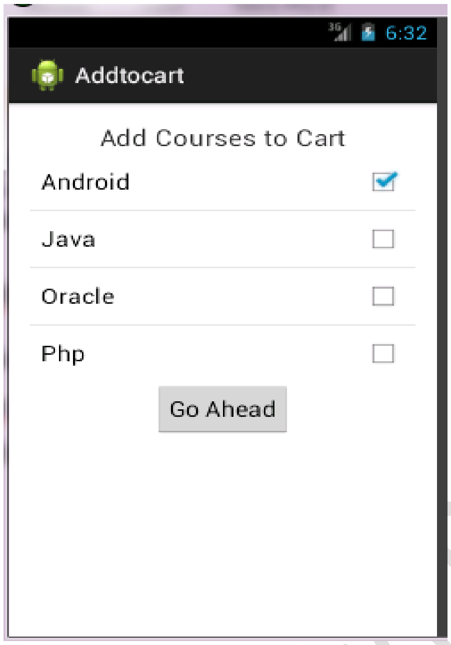
1. Main menu
2. Checkout
3. Add to cart
4. Exit

Main Menu Option

On clicking main menu option, control should go to **HomepageActv** (2nd Activity)

Add to cart Menu Option

On clicking Addtocart option, control should go to addtocartactv (7th activity). – refer figure 4



Add Courses to Cart

Android ☑

Java □

Oracle □

Php □

Figure 4

In this actv there is a listview containing list of courses with option of selection of course and one button to go ahead.

On clicking button, items in the List View should be written to a text File (say Mydata.txt)

Important tips for writing to File : Here first create a folder named “myfiles” inside this create a textfile named **Mydata.txt** .

// code snippet to fetch all the options selected in the List

```
b.setOnClickListener(new View.OnClickListener() {

@Override
public void onClick(View v) {
```

```
SparseBooleanArray check = list.getCheckedItemPositions();

ArrayList<String> aa = new ArrayList<String>();

for (int i = 0; i < check.size(); i++) {
        int pos = check.keyAt(i);
        if (check.valueAt(i)) {
        aa.add(AA.getItem(pos));
}
}
int n=list.getCheckedItemCount();



String f = "data.txt";
FileOutputStream fout = null;
try {
fout = openFileOutput(f, Context.MODE_PRIVATE);
OutputStreamWriter osw = new OutputStreamWriter(fout);

for (int k = 0; k < aa.size(); k++) {
osw.write(aa.get(k) + ",");
}
osw.close();
}
catch(….){…………}
```

> Exit menu Option

On clicking exit option, control should come out of the application. (refer code snippet at end of document)

> Checkout menu Option

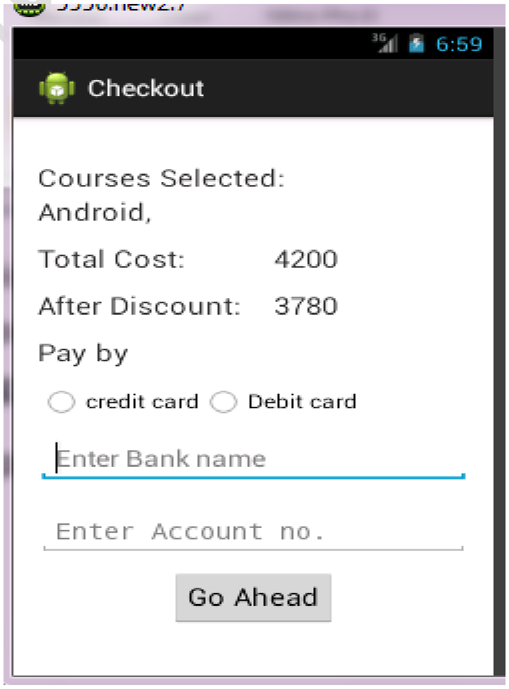
On clicking checkout option, call checkout Activity **(8th Activity)**.

In this checkout actv the details of courses selected, cost after discount should come in the textviews

The user should be able to enter

1. Mode of payment (using radiobuttons)
2. Bank name using Autocompletetextview
3. Password using EditText

See figure 5



Courses Selected:
Android,

Total Cost: 4200

After Discount: 3780

Pay by

○ credit card ○ Debit card

Enter Bank name

Enter Account no.

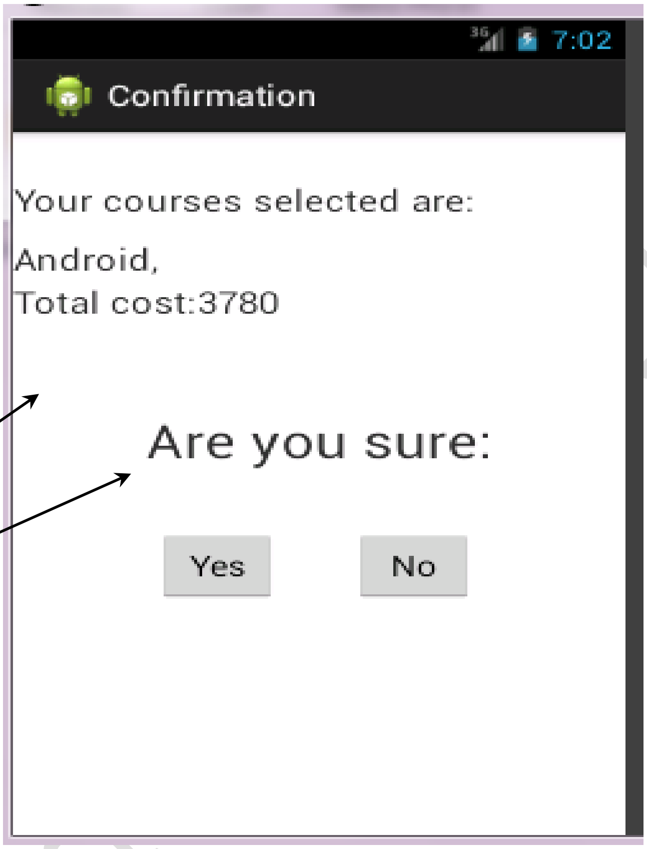
Figure 5

On clicking "go ahead" button , first check Bank name and Account Number has been filled. Just in case , if any field is blank , then show appropriate TOAST.

You should only move ahead, if all the Information is filled. **The data (courses and cost) is send via a Bundle Object to confirmation Activity.**

From **checkout actv control** should go to **confirmation Actv**

The confirmation activity **(9th Activity)** should show all courses (separated by commas) and discounts in 2 separate Text Views .



7:02

Confirmation

Your courses selected are:

Android,
Total cost:3780

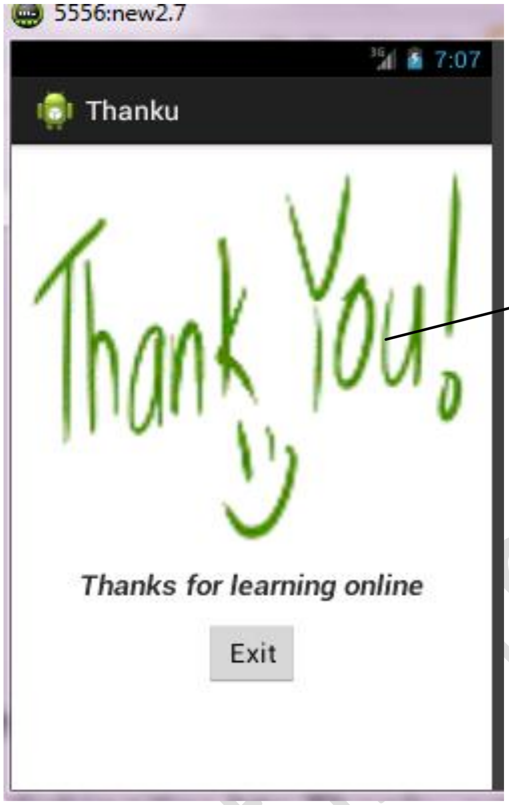
Are you sure:

Text View to show courses and total cost

Figure 6

In this, on clicking **"No"** button control should go to **addtocartactv** (**7th activity**).

On clicking "**Yes**" button call **ThankuActv** (refer figure 7). This is the last Activity **(10th Activity)**



Thanks for learning online

Figure 7

Can be **bg** Image or can show image in an **Image View**

On clicking **EXIT** button come out of the application.

Must read:

// sample code snippet - A – to be put in 3rd , 4th , 5th and 6th activity

```
case R.id.exit: // Exit menu option
Intent i4=new Intent(getBaseContext(),Homepage.class);
i4.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
i4.putExtra("exit", true);
startActivity(i4);
return true;
```

Explanation: Simply calling **finish() method would** only finish the current activity and won't finish the Application. Hence we set Intent.*FLAG_ACTIVITY_CLEAR_TOP* flag. This ensures that all the activities are cleared and control is transferred to HomePageActv. We put a Bundle with **key** as **exit** and **value** as **true , and pass it to** HomePageActv.

In the HomePageActv we must add the following code:

```
// sample code snippet - B
if(getIntent().getBooleanExtra("exit",false)){
    finish();
        }
```

Note that HomePage(i.e. the 2nd activity) which shows a list of all courses in a <TableLayout>, checks if "**exit**" key is set. If yes then it finishes itself.

This clearly indicates that if we have selected Exit option in the any of the files

- AndroidActv.java (3rd Activity)
- JavaActv.java (4th Activity)
- OracleActv.java (5th Activity)
- PHPActv.java (6th Activity)

Then we use **putExtra()** to set **"exit" key. (refer sample code – A)**

Then control goes back to HomePage Actv , which checks if Key **exit is set or not if yes then we exit the only actv "HomePageActv". (refer sample code – B).**