

```
1 package com.example.WaGbA;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import androidx.recyclerview.widget.
    LinearLayoutManager;
5
6 import android.content.Intent;
7 import android.os.Bundle;
8 import android.view.View;
9 import android.widget.ImageView;
10 import android.widget.TextView;
11 import android.widget.Toast;
12
13 import com.example.WaGbA.databinding.
    ActivityDishesBinding;
14 import com.firebase.ui.database.
    FirebaseRecyclerOptions;
15 import com.google.firebase.database.DatabaseReference
    ;
16 import com.google.firebase.database.FirebaseDatabase;
17
18 import java.text.SimpleDateFormat;
19 import java.util.Calendar;
20 import java.util.HashMap;
21 public class Dishes extends AppCompatActivity
    implements DishInterfaceClass {
22     private ActivityDishesBinding binding;
23     private DishesAdapter adapter;
24     private Intent receivePos;
25     DishesClass dishesClass;
26     String price;
27
28
29     @Override
30     protected void onCreate(Bundle savedInstanceState
    ) {
31         super.onCreate(savedInstanceState);
32         binding = ActivityDishesBinding.inflate(
    getLayoutInflater());
33         View view = binding.getRoot();
34         setContentView(view);
```

```

35         //img=findViewById(R.id.img1);
36         binding.recyclerview.setLayoutManager(new
    LinearLayoutManager(this));
37         receivePos=getIntent();
38         int position=receivePos.getIntExtra("pos",0);
39         switch (position){
40             case 0:
41                 FirebaseRecyclerOptions<DishesClass>kfc =
42                     new FirebaseRecyclerOptions.
    Builder<DishesClass>()
43                         .setQuery(
    FirebaseDatabase.getInstance().getReference().child("
    KFC"), DishesClass.class)
44                         .build();
45                 adapter = new DishesAdapter(kfc, this);
46                 binding.recyclerview.setAdapter(adapter);
47                 break;
48
49             case 1:
50                 FirebaseRecyclerOptions<DishesClass>
    kbb =
51                     new FirebaseRecyclerOptions.
    Builder<DishesClass>()
52                         .setQuery(
    FirebaseDatabase.getInstance().getReference().child("
    kabab"), DishesClass.class)
53                         .build();
54                 adapter = new DishesAdapter(kbb, (
    DishInterfaceClass) this);
55                 binding.recyclerview.setAdapter(adapter);
56                 break;
57
58             case 2:
59                 FirebaseRecyclerOptions<DishesClass>
    mc =
60                     new FirebaseRecyclerOptions.
    Builder<DishesClass>()
61                         .setQuery(
    FirebaseDatabase.getInstance().getReference().child("
    Mc"), DishesClass.class)
62                         .build();

```

```

63         adapter = new DishesAdapter(mc, (
        DishInterfaceClass) this);
64         binding.recyclerview.setAdapter(adapter);
65         break;
66
67         case 3:
68             FirebaseRecyclerOptions<DishesClass>
        pg=
69                 new FirebaseRecyclerOptions.
        Builder<DishesClass>()
70                     .setQuery(
        FirebaseDatabase.getInstance().getReference().child(
        "pergo"), DishesClass.class)
71                     .build();
72             adapter = new DishesAdapter(pg, (
        DishInterfaceClass) this);
73             binding.recyclerview.setAdapter(adapter);
74             break;
75
76         case 4:
77             FirebaseRecyclerOptions<DishesClass>
        bk =
78                 new FirebaseRecyclerOptions.
        Builder<DishesClass>()
79                     .setQuery(
        FirebaseDatabase.getInstance().getReference().child(
        "bkry"), DishesClass.class)
80                     .build();
81             adapter = new DishesAdapter(bk, (
        DishInterfaceClass) this);
82             binding.recyclerview.setAdapter(adapter);
83             break;
84
85         case 5:
86             FirebaseRecyclerOptions<DishesClass>
        co =
87                 new FirebaseRecyclerOptions.
        Builder<DishesClass>()
88                     .setQuery(
        FirebaseDatabase.getInstance().getReference().child(
        "coash"), DishesClass.class)

```

```

89                                     .build();
90         adapter = new DishesAdapter(co, (
    DishInterfaceClass) this);
91         binding.recyclerview.setAdapter(adapter);
92         break;
93
94         case 6:
95             FirebaseRecyclerOptions<DishesClass>
    spicy=
96                 new FirebaseRecyclerOptions.
    Builder<DishesClass>()
97                     .setQuery(
    FirebaseDatabase.getInstance().getReference().child(
    "Spicy"), DishesClass.class)
98                     .build();
99             adapter = new DishesAdapter(spicy, (
    DishInterfaceClass) this);
100            binding.recyclerview.setAdapter(adapter);
101            break;
102
103            case 7:
104                FirebaseRecyclerOptions<DishesClass>
    ind =
105                    new FirebaseRecyclerOptions.
    Builder<DishesClass>()
106                        .setQuery(
    FirebaseDatabase.getInstance().getReference().child(
    "hindi"), DishesClass.class)
107                        .build();
108                adapter = new DishesAdapter(ind, (
    DishInterfaceClass) this);
109                binding.recyclerview.setAdapter(adapter);
110                break;
111
112
113            case 8:
114                FirebaseRecyclerOptions<DishesClass>
    tom=
115                    new FirebaseRecyclerOptions.
    Builder<DishesClass>()
116                        .setQuery(

```

```

116 FirebaseDatabase.getInstance().getReference().child(
    "Koshry"), DishesClass.class)
117                                     .build();
118         adapter = new DishesAdapter(tom, (
    DishInterfaceClass) this);
119         binding.recyclerview.setAdapter(adapter);
120         break;
121
122         case 9:
123             FirebaseRecyclerOptions<DishesClass>
    fsh=
124                 new FirebaseRecyclerOptions.
    Builder<DishesClass>()
125                     .setQuery(
    FirebaseDatabase.getInstance().getReference().child(
    "fish"), DishesClass.class)
126                     .build();
127             adapter = new DishesAdapter(fsh, (
    DishInterfaceClass) this);
128             binding.recyclerview.setAdapter(adapter);
129             break;
130
131     }
132
133 }
134
135 @Override
136 protected void onStart() {
137     super.onStart();
138     adapter.startListening();
139 }
140
141 @Override
142 protected void onStop() {
143     super.onStop();
144     adapter.stopListening();
145 }
146
147 @Override
148 public void onItemClick(int position) {
149     Toast.makeText(Dishes.this, "Added to your

```

```
149 cart", Toast.LENGTH_LONG).show();  
150  
151     }  
152  
153     }  
154
```

```
1 package com.example.WaGbA;
2
3 public class Orders {
4
5     private String MealDetails, OrdeSstatus, Quantity;
6
7     public Orders(){}
8
9     public Orders(String mealDetails, String
    ordeSstatus, String quantity) {
10         MealDetails = mealDetails;
11         OrdeSstatus = ordeSstatus;
12         Quantity = quantity;
13     }
14
15     public String getMealDetails() {
16         return MealDetails;
17     }
18
19     public void setMealDetails(String mealDetails) {
20         MealDetails = mealDetails;
21     }
22
23     public String getOrdeSstatus() {
24         return OrdeSstatus;
25     }
26
27     public void setOrdeSstatus(String ordeSstatus) {
28         OrdeSstatus = ordeSstatus;
29     }
30
31     public String getQuantity() {
32         return Quantity;
33     }
34
35     public void setQuantity(String quantity) {
36         Quantity = quantity;
37     }
38 }
39
```

```
1 package com.example.WaGbA;
2
3 import androidx.annotation.NonNull;
4 import androidx.appcompat.app.AppCompatActivity;
5
6 import android.content.Intent;
7 import android.os.Bundle;
8 import android.util.Patterns;
9 import android.view.View;
10 import android.widget.Toast;
11
12 import com.example.WaGbA.databinding.
    ActivitySignInBinding;
13 import com.google.android.gms.tasks.
    OnCompleteListener;
14 import com.google.android.gms.tasks.Task;
15 import com.google.firebase.auth.AuthResult;
16 import com.google.firebase.auth.FirebaseAuth;
17
18 import java.util.regex.Pattern;
19
20 public class SignIn extends AppCompatActivity {
21
22     private ActivitySignInBinding binding;
23     private FirebaseAuth mAuth;
24     private Intent mainPage;
25     public static final Pattern School_Email
26         = Pattern.compile(
27         "[a-zA-Z0-9\\+\\.\\_\\%\\-\\+]{1,256}" +
28         "\\@" + "eng.asu.edu.eg"
29     );
30
31     @Override
32     protected void onCreate(Bundle savedInstanceState
33     ) {
34         super.onCreate(savedInstanceState);
35         binding = ActivitySignInBinding.inflate(
36         getLayoutInflater());
37         View view = binding.getRoot();
38         setContentView(view);
```



```
38
39         mAuth = FirebaseAuth.getInstance();
40         mainPage=new Intent(this, RestaurantsPage.
class);
41         binding.createbtn.setOnClickListener(new View
.OnClickListener() {
42             @Override
43             public void onClick(View view) {
44
45                 createAccount();
46
47             }
48         });
49
50     }
51
52     private void createAccount() {
53
54         String name=binding.nameEdt.getText().
toString();
55         String mail=binding.mailEdt.getText().
toString();
56         String phone=binding.phoneEdt.getText().
toString();
57         String address=binding.addrEdt.getText().
toString();
58         String password=binding.passEdt.getText().
toString();
59
60         if(name.isEmpty()){
61             binding.nameEdt.setError("Name is
required!");
62             binding.nameEdt.requestFocus();
63             return;
64         }
65         if(mail.isEmpty()){
66             binding.mailEdt.setError("Email is
required!");
67             binding.mailEdt.requestFocus();
68             return;
69         }
```

```
70
71         if(!Patterns.PHONE.matcher(phone).matches
72         ()){
73             binding.phoneEdt.setError("Please
74             provide a valid phone number!");
75             binding.phoneEdt.requestFocus();
76             return;
77         }
78         if(address.isEmpty()){
79             binding.addrEdt.setError("Address is
80             required!");
81             binding.addrEdt.requestFocus();
82             return;
83         }
84         if(password.isEmpty()){
85             binding.passEdt.setError("Password is
86             required!");
87             binding.passEdt.requestFocus();
88             return;
89         }
90         if(password.length()<6){
91             binding.passEdt.setError("Password
92             length must be >=6 characters!");
93             binding.passEdt.requestFocus();
94             return;
95         }
96         if(!School_Email.matcher(mail).matches()){
97             binding.mailEdt.setError("Please provide
98             a valid email address!");
99             binding.mailEdt.requestFocus();
100             return;
101         }
102         UserEntity userEntity=new UserEntity();
103         userEntity.setUsername(binding.nameEdt.
104         getText().toString());
105         userEntity.setUsermail(binding.mailEdt.
106         getText().toString());
```

```

103         userEntity.setUserphone(Integer.parseInt(
            binding.phoneEdt.getText().toString()));
104         userEntity.setUseraddress(binding.addrEdt.
            getText().toString());
105         userEntity.setUserpassword(binding.passEdt.
            getText().toString());
106
107         UserDatabase userDatabase=UserDatabase.
            getUserDatabase(getApplicationContext());
108         UserDao userDao=userDatabase.userDao();
109         runOnUiThread(new Runnable() {
110             @Override
111             public void run() {
112
113             }
114         });
115
116         mAuth.createUserWithEmailAndPassword(
            mail, password)
117             .addOnCompleteListener(new
            OnCompleteListener<AuthResult>() {
118                 @Override
119                 public void onComplete(@
            NonNull Task<AuthResult> task) {
120                     if (task.isSuccessful
            ()) {
121                         Toast.makeText(
            SignIn.this, "Successfully Signed In!", Toast.
            LENGTH_LONG).show();
122                         startActivity(
            mainPage);
123                         finish();
124                     } else {
125                         // If sign in fails
, display a message to the user.
126                         Toast.makeText(
            SignIn.this, "Sign in failed!Make another trial!",
            Toast.LENGTH_LONG).show();
127
128                     }
129                 }

```

```
130         });  
131     }  
132 }  
133  
134
```

```
1 package com.example.WaGbA;
2
3 import androidx.room.Dao;
4 import androidx.room.Insert;
5
6 @Dao
7 public interface UserDao {
8     @Insert
9     void registerUser(UserEntity userEntity);
10 }
11
```

```
1 package com.example.WaGbA;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.annotation.SuppressLint;
6 import android.content.Intent;
7 import android.os.Bundle;
8 import android.util.Log;
9 import android.view.Display;
10 import android.view.View;
11 import android.widget.Button;
12 import android.widget.EditText;
13 import android.widget.ImageView;
14 import android.widget.TextView;
15 import android.widget.Toast;
16
17 import com.bumptech.glide.Glide;
18 import com.cepheuen.elegantnumberbutton.view.
    ElegantNumberButton;
19 import com.firebase.ui.database.
    FirebaseRecyclerOptions;
20 import com.google.firebase.database.DatabaseReference
    ;
21 import com.google.firebase.database.FirebaseDatabase;
22
23 import java.util.HashMap;
24
25 public class CartPage extends AppCompatActivity {
26
27     private TextView totPrice, totAmount, Delivery,
    Display,availability;
28     private EditText userID;
29     private ImageView img;
30     private Intent receivePrice, receiveImg;
31     private Button confirm, remove, add, sub, cash, credit
    ;
32     private int p, totalQuantity=1;
33     private FirebaseDatabase firebaseDatabase;
34     private DatabaseReference databaseReference;
35     private HashMap<String ,Object> orders;
36     private String pMethod;
```

```

37
38
39
40
41
42
43     @SuppressWarnings("MissingInflatedId")
44     @Override
45     protected void onCreate(Bundle savedInstanceState
46 ) {
47         super.onCreate(savedInstanceState);
48         setContentView(R.layout.activity_cart_page);
49         firebaseDatabase=FirebaseDatabase.getInstance
50 ();
51         databaseReference=firebaseDatabase.
52 getReference().child("Orders").child("Order Details"
53 );
54
55         img=findViewById(R.id.img1);
56         totPrice=findViewById(R.id.tPrice);
57         totAmount=findViewById(R.id.total);
58         Delivery=findViewById(R.id.delivery);
59         add=findViewById(R.id.addbtn);
60         sub=findViewById(R.id.subbtn);
61         userID=findViewById(R.id.userId);
62         availability=findViewById(R.id.availTv);
63         cash=findViewById(R.id.cash);
64         credit=findViewById(R.id.credit);
65
66         Display=findViewById(R.id.displayedt);
67         receivePrice=getIntent();
68         receiveImg=getIntent();
69         String url=receiveImg.getStringExtra("
70 orderImg");
71         Glide.with(this).load(url).into(img);
72         String price = receivePrice.getStringExtra("
73 price");
74
75         totPrice.setText(price);

```

```

72         p= Integer.parseInt(totPrice.getText().
    toString());
73         totAmount.setText(String.valueOf(p*
    totalQuantity));
74
75         cash.setOnClickListener(new View.
    OnClickListener() {
76             @Override
77             public void onClick(View v) {
78                 pMethod="Cash";
79
80             }
81         });
82         credit.setOnClickListener(new View.
    OnClickListener() {
83             @Override
84             public void onClick(View v) {
85                 pMethod="Credit";
86             }
87         });
88
89         add.setOnClickListener(new View.
    OnClickListener() {
90             @Override
91             public void onClick(View v) {
92                 if(totalQuantity<100) {
93                     totalQuantity++;
94                     Display.setText(String.valueOf(
    totalQuantity));
95                     totAmount.setText(String.valueOf(p
    *totalQuantity));
96                     if(totalQuantity>=10){
97                         availability.setText("
    unavailable");
98                     }
99
100                }
101            }
102        });
103
104        sub.setOnClickListener(new View.

```



```

104 OnClickListener() {
105     @Override
106     public void onClick(View v) {
107         if (totalQuantity > 1) {
108             totalQuantity--;
109             Display.setText(String.valueOf(
totalQuantity));
110             totAmount.setText(String.valueOf(p
* totalQuantity));
111             if(totalQuantity<=10){
112                 availability.setText("
available");
113             }
114         }
115     }
116 });
117
118
119
120     confirm=findViewById(R.id.confirm);
121     confirm.setOnClickListener(new View.
OnClickListener() {
122         @Override
123         public void onClick(View v){
124
125             if(availability.getText().toString
()=="unavailable"){
126                 availability.setError("we have
only 10 dishes of this meal!");
127                 availability.requestFocus();
128                 return;
129
130             }else {
131
132                 Intent OrderStatIntent = new
Intent(CartPage.this, OrderStatus.class);
133                 orders = new HashMap<>();
134                 orders.put("meal details",url);
135                 orders.put("Quantity",
totalQuantity);
136                 orders.put("payment method",

```

```

136 pMethod);
137         String msg = "Waiting for
        acceptance";
138         orders.put("Order status", msg);
139         if (userID.getText().toString().
        isEmpty()) {
140             userID.setError("your id is
        required to track your order!");
141             userID.requestFocus();
142             return;
143
144         } else if (userID.length() < 7
        ) {
145             userID.setError("your id
        must be<=7!");
146             userID.requestFocus();
147             return;
148
149         } else {
150             orders.put("UserID", userID.
        getText().toString());
151         }
152         databaseReference.setValue(
        orders);
153         startActivity(OrderStatIntent);
154     }
155 }
156 });
157
158     remove=findViewById(R.id.Remove);
159     remove.setOnClickListener(new View.
        OnClickListener() {
160         @Override
161         public void onClick(View v) {
162
163             databaseReference.setValue(null);
164             Intent intent=new Intent(CartPage.
        this,Dishes.class);
165             startActivity(intent);
166
167         }

```

```
168         });  
169  
170  
171  
172     }  
173 }
```

```
1 package com.example.WaGbA;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.content.Intent;
6 import android.os.Bundle;
7 import android.view.View;
8
9 import com.example.WaGbA.databinding.
    ActivityHomePageBinding;
10
11 public class HomePage extends AppCompatActivity {
12
13     private ActivityHomePageBinding binding;
14     Intent loginpage,signpage;
15
16     @Override
17     protected void onCreate(Bundle savedInstanceState
18 ) {
19         super.onCreate(savedInstanceState);
20         binding = ActivityHomePageBinding.inflate(
21             getLayoutInflater());
22         View view = binding.getRoot();
23         setContentView(view);
24         loginpage=new Intent(this,LoginPage.class);
25         signpage=new Intent(this,SignIn.class);
26
27         binding.logbtn.setOnClickListener(new View.
28             OnClickListener() {
29                 @Override
30                 public void onClick(View view) {
31                     startActivity(loginpage);
32                 }
33             });
34         binding.signbtn.setOnClickListener(new View.
35             OnClickListener() {
36                 @Override
37                 public void onClick(View view) {
38                     startActivity(signpage);
```

```
37
38         }
39     });
40 }
41 }
```

```
1 package com.example.WaGbA;
2
3 import androidx.annotation.NonNull;
4 import androidx.appcompat.app.AppCompatActivity;
5
6 import android.content.Intent;
7 import android.os.Bundle;
8 import android.view.View;
9 import android.widget.Toast;
10
11 import com.example.WaGbA.databinding.
    ActivityLoginPageBinding;
12 import com.google.android.gms.tasks.
    OnCompleteListener;
13 import com.google.android.gms.tasks.Task;
14 import com.google.firebase.auth.AuthResult;
15 import com.google.firebase.auth.FirebaseAuth;
16
17 public class LoginPage extends AppCompatActivity {
18     private FirebaseAuth mAuth;
19     private ActivityLoginPageBinding binding;
20     Intent mainPage;
21     @Override
22     protected void onCreate(Bundle savedInstanceState
    ) {
23         super.onCreate(savedInstanceState);
24         binding = ActivityLoginPageBinding.inflate(
    getLayoutInflater());
25         View view = binding.getRoot();
26         setContentView(view);
27         mAuth = FirebaseAuth.getInstance();
28         mainPage=new Intent(this, RestaurantsPage.
    class);
29
30         binding.loginbtn.setOnClickListener(new View.
    OnClickListener() {
31             @Override
32             public void onClick(View view) {
33                 binding.loginbtn.setOnClickListener(
    new View.OnClickListener() {
34                     @Override
```

```
35         public void onClick(View view) {
36             loginAccount();
37         }
38     });
39
40     }
41     });
42 }
43
44     private void loginAccount() {
45
46         String mail = binding.mailtxt.getText().
47         toString();
48         String password = binding.passwordtxt.getText
49         ().toString();
50
51         if (mail.isEmpty()) {
52             binding.mailtxt.setError("Email is
53             required!");
54             binding.mailtxt.requestFocus();
55             return;
56         }
57         if (password.isEmpty()) {
58             binding.passwordtxt.setError("Password is
59             required!");
60             binding.passwordtxt.requestFocus();
61             return;
62         }
63
64         mAuth.signInWithEmailAndPassword(mail,
65         password)
66         .addOnCompleteListener(new
67         OnCompleteListener<AuthResult>() {
68             @Override
69             public void onComplete(@NonNull
70             Task<AuthResult> task) {
71                 if (task.isSuccessful()) {
72                     Toast.makeText(LoginPage.
73                     this, "Logged in successfully!", Toast.LENGTH_LONG).
74                     show();
75
76                     startActivity(mainPage);
77                 }
78             }
79         });
80     }
81 }
```

```
67         finish();
68
69
70     } else {
71         // If sign in fails,
72         display a message to the user.
73         Toast.makeText(LoginPage
74             .this,
75             "Email or
76             Password is incorrect please try again", Toast.
77             LENGTH_LONG).show();
78
79     }
80
81 }
82 }
```



```
1 package com.example.WaGbA;
2
3 import androidx.room.ColumnInfo;
4 import androidx.room.Entity;
5 import androidx.room.PrimaryKey;
6
7 @Entity(tableName = "user")
8 public class UserEntity {
9     @PrimaryKey(autoGenerate = true)
10     int id;
11
12     @ColumnInfo(name = "username")
13     String username;
14
15     @ColumnInfo(name = "usermail")
16     String usermail;
17
18     @ColumnInfo(name = "userPhone")
19     int userphone;
20
21     @ColumnInfo(name = "useraddress")
22     String useraddress;
23
24     @ColumnInfo(name = "userpassword")
25     String userpassword;
26
27     public int getId() {
28         return id;
29     }
30
31     public void setId(int id) {
32         this.id = id;
33     }
34
35     public String getUsername() {
36         return username;
37     }
38
39     public void setUsername(String username) {
40         this.username = username;
41     }
```

```
42
43     public String getUsermail() {
44         return usermail;
45     }
46
47     public void setUsermail(String usermail) {
48         this.usermail = usermail;
49     }
50
51     public int getUserphone() {
52         return userphone;
53     }
54
55     public void setUserphone(int userphone) {
56         this.userphone = userphone;
57     }
58
59     public String getUseraddress() {
60         return useraddress;
61     }
62
63     public void setUseraddress(String useraddress) {
64         this.useraddress = useraddress;
65     }
66
67     public String getUserpassword() {
68         return userpassword;
69     }
70
71     public void setUserpassword(String userpassword
72 ) {
73         this.userpassword = userpassword;
74     }
75 }
```

```
1 package com.example.WaGbA;
2
3 public class DishesClass {
4
5     String purl,price;
6     DishesClass()
7     {
8
9     }
10
11     public DishesClass(String purl, String price) {
12         this.purl = purl;
13         this.price = price;
14     }
15
16     public String getPrice() {
17         return price;
18     }
19
20     public void setPrice(String price) {
21         this.price = price;
22     }
23
24     public String getPurl() {
25         return purl;
26     }
27
28     public void setPurl(String purl) {
29         this.purl = purl;
30     }
31 }
```

```
1 package com.example.WaGbA;
2
3 import androidx.annotation.NonNull;
4 import androidx.appcompat.app.AppCompatActivity;
5
6 import android.annotation.SuppressLint;
7 import android.content.Intent;
8 import android.os.Bundle;
9 import android.view.View;
10 import android.widget.Button;
11 import android.widget.EditText;
12 import android.widget.TextView;
13
14 import com.google.firebase.database.DataSnapshot;
15 import com.google.firebase.database.DatabaseError;
16 import com.google.firebase.database.DatabaseReference
17 ;
18 import com.google.firebase.database.FirebaseDatabase;
19 import com.google.firebase.database.
20 ValueEventListener;
21
22 public class OrderStatus extends AppCompatActivity {
23     private TextView accept;
24     private FirebaseDatabase firebaseD;
25     private DatabaseReference readData;
26     private EditText UsrMail;
27     private Button lsbtn,Restaurant;
28
29     @SuppressWarnings("MissingInflatedId")
30     @Override
31     protected void onCreate(Bundle savedInstanceState
32 ) {
33         super.onCreate(savedInstanceState);
34         setContentView(R.layout.activity_order_status
35 );
36
37         UsrMail=findViewById(R.id.usrml);
38         firebaseD=FirebaseDatabase.getInstance();
39         accept=findViewById(R.id.confTv);
40         lsbtn=findViewById(R.id.lst);
```

```

38         Restaurant=findViewById(R.id.button);
39         Restaurant.setOnClickListener(new View.
    OnClickListener() {
40             @Override
41             public void onClick(View v) {
42                 Intent intent=new Intent(OrderStatus
    .this,RestaurantsPage.class);
43                 startActivity(intent);
44             }
45         });
46         firebaseD=FirebaseDatabase.getInstance();
47         lsbtn.setOnClickListener(new View.
    OnClickListener() {
48             @Override
49             public void onClick(View v) {
50
51
52
53                 if(!UsrMail.getText().toString().
    isEmpty()) {
54
55                     readData=firebaseD.getReference
    ().child("Accepted Orders")
56                     .child(UsrMail.getText().
    toString());
57
58                     readData.
    addListenerForSingleValueEvent(new ValueEventListener
    () {
59                         @Override
60                         public void onDataChange(@
    NonNull DataSnapshot snapshot) {
61                             String val = snapshot.
    getValue().toString();
62                             accept.setText(val);
63                         }
64
65                         @Override
66                         public void onCancelled(@
    NonNull DatabaseError error) {
67

```

```
68         }
69         });
70         }else { accept.setText(" ");
71
72         }
73     }
74     });
75
76 }
77
78 }
```

```
1 package com.example.WaGbA;
2
3 import android.content.Context;
4
5 import androidx.room.Database;
6 import androidx.room.Room;
7 import androidx.room.RoomDatabase;
8
9 @Database(entities = {UserEntity.class}, version = 1)
10
11 public abstract class UserDatabase extends
    RoomDatabase {
12     private static final String dbName="user";
13     private static UserDatabase userDatabase;
14     public static synchronized UserDatabase
    getUserDatabase(Context context){
15
16         if(userDatabase==null){
17             userDatabase= Room.databaseBuilder(
    context,UserDatabase.class,dbName).
    fallbackToDestructiveMigration().build();
18         }
19         return userDatabase;
20     }
21
22     public abstract UserDao userDao();
23 }
24
```

```
1  package com.example.WaGbA;
2
3  import androidx.annotation.NonNull;
4  import androidx.recyclerview.widget.RecyclerView;
5
6  import android.content.Intent;
7  import android.view.LayoutInflater;
8  import android.view.View;
9  import android.view.ViewGroup;
10 import android.widget.ImageView;
11 import android.widget.TextView;
12
13 import com.bumptech.glide.Glide;
14 import com.firebase.ui.database.
    FirebaseRecyclerAdapter;
15 import com.firebase.ui.database.
    FirebaseRecyclerOptions;
16
17 public class DishesAdapter extends
    FirebaseRecyclerAdapter<DishesClass,DishesAdapter.
    myviewholder> {
18
19     private final DishInterfaceClass
    dishInterfaceClass;
20
21     public DishesAdapter(@NonNull
    FirebaseRecyclerOptions<DishesClass> options,
    DishInterfaceClass dishInterfaceClass) {
22         super(options);
23         this.dishInterfaceClass = dishInterfaceClass
    ;
24     }
25
26     @Override
27     protected void onBindViewHolder(@NonNull
    myviewholder holder, int position, @NonNull
    DishesClass DishesClass) {
28         holder.pricetxt.setText(DishesClass.getPrice
    ());
29         Glide.with(holder.img.getContext()).load(
    DishesClass.getPurl()).into(holder.img);
```



```

30         holder.itemView.setOnClickListener(new View.
    OnClickListener() {
31             @Override
32             public void onClick(View v) {
33                 Intent intent=new Intent(holder.img.
    getContext(),CartPage.class);
34                 intent.putExtra("price",DishesClass.
    getPrice());
35                 intent.putExtra("orderImg",
    DishesClass.getPurl());
36                 holder.img.getContext().
    startActivity(intent);
37             }
38         });
39     }
40 }
41
42 @NonNull
43 @Override
44 public myviewholder onCreateViewHolder(@NonNull
    ViewGroup parent, int viewType) {
45     View view= LayoutInflater.from(parent.
    getContext()).inflate(R.layout.dishes_item,parent,
    false);
46     return new myviewholder(view,
    dishInterfaceClass);
47 }
48
49 class myviewholder extends RecyclerView.
    ViewHolder {
50     ImageView img;
51     TextView pricetxt;
52     public myviewholder(@NonNull View itemView,
    DishInterfaceClass dishInterfaceClass)
53     {
54         super(itemView);
55         img=itemView.findViewById(R.id.img1);
56         pricetxt=(TextView)itemView.findViewById(
    R.id.priceTxt);
57
58         itemView.setOnClickListener(new View.

```

```
58 OnClickListener() {
59     @Override
60     public void onClick(View v) {
61         if( DishesAdapter.this.
dishInterfaceClass !=null){
62             int pos= getAdapterPosition
63             ();
64             if(pos!=RecyclerView.
NO_POSITION){
65                 DishesAdapter.this.
dishInterfaceClass.onClick(pos);
66             }
67         }
68     }
69 });
70
71 }
72 }
73
74 }
75
76
```

```

1 package com.example.WaGbA;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import androidx.recyclerview.widget.
  LinearLayoutManager;
5
6 import android.content.Intent;
7 import android.os.Bundle;
8 import android.view.View;
9 import android.widget.Button;
10
11
12 import com.example.WaGbA.databinding.
  ActivityRestaurantsPageBinding;
13 import com.firebase.ui.database.
  FirebaseRecyclerOptions;
14 import com.google.firebase.database.FirebaseDatabase;
15
16 public class RestaurantsPage extends
  AppCompatActivity implements DishInterfaceClass
17 {
18     private ActivityRestaurantsPageBinding binding;
19     private RestaurantsAdapter adapter;
20     private Intent orderIntent, CartIntent;
21     private Button Order;
22
23     @Override
24     protected void onCreate(Bundle savedInstanceState) {
25         super.onCreate(savedInstanceState);
26         binding = ActivityRestaurantsPageBinding .
  inflate(getLayoutInflater());
27         View view = binding.getRoot();
28         setContentView(view);
29         binding.recyclerview.setLayoutManager(new
  LinearLayoutManager(this));
30         Order=findViewById(R.id.orderPagebtn);
31         Order.setOnClickListener(new View.
  OnClickListener() {
32             @Override
33             public void onClick(View v) {

```

```

34             Intent intent=new Intent(
        RestaurantsPage.this,OrderStatus.class);
35             startActivity(intent);
36         }
37     });
38
39
40
41     FirebaseRecyclerOptions<RestaurantsClass>
options =
42         new FirebaseRecyclerOptions.Builder<
        RestaurantsClass>()
43             .setQuery(FirebaseDatabase.
        getInstance().getReference().child("Restaurants"),
        RestaurantsClass.class)
44             .build();
45
46     adapter=new RestaurantsAdapter(options, this
    );
47     binding.recyclerview.setAdapter(adapter);
48
49 }
50
51 @Override
52 protected void onStart() {
53     super.onStart();
54     adapter.startListening();
55 }
56
57 @Override
58 protected void onStop() {
59     super.onStop();
60     adapter.stopListening();
61 }
62
63 @Override
64 public void onItemClick(int position) {
65     Intent intent=new Intent(RestaurantsPage.this
    ,Dishes.class);
66     intent.putExtra("pos",position);
67     startActivity(intent);

```

```
68         }  
69     }  
70
```

```
1 package com.example.WaGbA;
2
3 public class RestaurantsClass
4 {
5     String name,purl;
6     RestaurantsClass()
7     {
8
9     }
10    public RestaurantsClass(String name, String purl
11    ) {
12        this.name = name;
13        this.purl = purl;
14    }
15    public String getName() {
16        return name;
17    }
18
19    public void setName(String name) {
20        this.name = name;
21    }
22
23
24    public String getPurl() {
25        return purl;
26    }
27
28    public void setPurl(String purl) {
29        this.purl = purl;
30    }
31 }
32
```

```
1 package com.example.WaGbA;  
2  
3 public interface DishInterfaceClass {  
4     void onItemClick(int position );  
5 }  
6
```

```
1 package com.example.WaGbA;
2
3 import android.view.LayoutInflater;
4 import android.view.View;
5 import android.view.ViewGroup;
6 import android.widget.ImageView;
7 import android.widget.TextView;
8
9 import androidx.annotation.NonNull;
10 import androidx.recyclerview.widget.RecyclerView;
11
12 import com.bumptech.glide.Glide;
13 import com.firebase.ui.database.
    FirebaseRecyclerAdapter;
14 import com.firebase.ui.database.
    FirebaseRecyclerOptions;
15
16 public class RestaurantsAdapter extends
    FirebaseRecyclerAdapter<RestaurantsClass,
    RestaurantsAdapter.myviewholder>
17 {
18     private final DishInterfaceClass
    dishInterfaceClass;
19
20     public RestaurantsAdapter(@NonNull
    FirebaseRecyclerOptions<RestaurantsClass> options,
    DishInterfaceClass dishInterfaceClass) {
21         super(options);
22         this.dishInterfaceClass = dishInterfaceClass;
23     }
24
25     @Override
26     protected void onBindViewHolder(@NonNull
    myviewholder holder, int position, @NonNull
    RestaurantsClass RestaurantsClass)
27     {
28         holder.name.setText(RestaurantsClass.getName
    ());
29         Glide.with(holder.img.getContext()).load(
    RestaurantsClass.getPurl()).into(holder.img);
30     }
```



```

31
32     @NonNull
33     @Override
34     public myviewholder onCreateViewHolder(@NonNull
ViewGroup parent, int viewType)
35     {
36         View view= LayoutInflater.from(parent.
getContext()).inflate(R.layout.restaurants_iteem,
parent, false);
37         return new myviewholder(view,
dishInterfaceClass);
38     }
39
40     class myviewholder extends RecyclerView.
ViewHolder
41     {
42         ImageView img;
43         TextView name;
44         public myviewholder(@NonNull View itemView,
DishInterfaceClass dishInterfaceClass)
45         {
46             super(itemView);
47             img=itemView.findViewById(R.id.img1);
48             name=(TextView)itemView.findViewById(R.id
.nametext);
49
50             itemView.setOnClickListener(new View.
OnClickListener() {
51                 @Override
52                 public void onClick(View v) {
53                     if( RestaurantsAdapter.this.
dishInterfaceClass !=null){
54                         int pos= getAdapterPosition
();
55
56                         if(pos!=RecyclerView.
NO_POSITION){
57                             RestaurantsAdapter.this.
dishInterfaceClass.onItemClick(pos);
58                     }
59                 }

```

```
60         }
61     });
62
63     }
64 }
65 }
66
```