```java
package com.example.WaGbA;

import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.WaGbA.databinding.ActivityDishesBinding;
import com.firebase.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.HashMap;
public class Dishes extends AppCompatActivity implements DishInterfaceClass  {
    private ActivityDishesBinding binding;
    private DishesAdapter adapter;
    private Intent receivePos;
    DishesClass dishesClass;
    String price;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding = ActivityDishesBinding.inflate(getLayoutInflater());
        View view = binding.getRoot();
        setContentView(view);
        //img=findViewById(R.id.img1);
        binding.recview.setLayoutManager(new LinearLayoutManager(this));
        receivePos=getIntent();
        int position=receivePos.getIntExtra("pos",0);
        switch (position){
            case 0:
            FirebaseRecyclerOptions<DishesClass>kfc =
                    new FirebaseRecyclerOptions.Builder<DishesClass>()
                            .setQuery(FirebaseDatabase.getInstance().getReference().child("KFC"), DishesClass.class)
                            .build();
            adapter = new DishesAdapter(kfc, this);
            binding.recview.setAdapter(adapter);
            break;

            case 1:
                FirebaseRecyclerOptions<DishesClass> kbb =
                        new FirebaseRecyclerOptions.Builder<DishesClass>()
                                .setQuery(FirebaseDatabase.getInstance().getReference().child("kabab"), DishesClass.class)
                                .build();
                adapter = new DishesAdapter(kbb, (DishInterfaceClass) this);
                binding.recview.setAdapter(adapter);
                break;

            case 2:
                FirebaseRecyclerOptions<DishesClass> mc =
                        new FirebaseRecyclerOptions.Builder<DishesClass>()
                                .setQuery(FirebaseDatabase.getInstance().getReference().child("Mc"), DishesClass.class)
                                .build();
                 adapter = new DishesAdapter(mc, (DishInterfaceClass) this);
                 binding.recview.setAdapter(adapter);
                 break;

             case 3:
                 FirebaseRecyclerOptions<DishesClass> pg=
                         new FirebaseRecyclerOptions.Builder<DishesClass>()
                                 .setQuery(FirebaseDatabase.getInstance().getReference().child("pergo"), DishesClass.class)
                                 .build();
                 adapter = new DishesAdapter(pg,(DishInterfaceClass) this);
                 binding.recview.setAdapter(adapter);
                 break;

             case 4:
                 FirebaseRecyclerOptions<DishesClass> bk =
                         new FirebaseRecyclerOptions.Builder<DishesClass>()
                                 .setQuery(FirebaseDatabase.getInstance().getReference().child("bkry"), DishesClass.class)
                                 .build();
                 adapter = new DishesAdapter(bk,(DishInterfaceClass) this);
                 binding.recview.setAdapter(adapter);
                 break;

             case 5:
                 FirebaseRecyclerOptions<DishesClass> co =
                         new FirebaseRecyclerOptions.Builder<DishesClass>()
                                 .setQuery(FirebaseDatabase.getInstance().getReference().child("coash"), DishesClass.class)
                                 .build();
                 adapter = new DishesAdapter(co, (DishInterfaceClass) this);
                 binding.recview.setAdapter(adapter);
                 break;

             case 6:
                 FirebaseRecyclerOptions<DishesClass> spicy=
                     new FirebaseRecyclerOptions.Builder<DishesClass>()
                                 .setQuery(FirebaseDatabase.getInstance().getReference().child("Spicy"), DishesClass.class)
                                 .build();
                 adapter = new DishesAdapter(spicy, (DishInterfaceClass) this);
                 binding.recview.setAdapter(adapter);
                 break;

             case 7:
                 FirebaseRecyclerOptions<DishesClass> ind =
                     new FirebaseRecyclerOptions.Builder<DishesClass>()
                                 .setQuery(FirebaseDatabase.getInstance().getReference().child("hindi"), DishesClass.class)
                                 .build();
                 adapter = new DishesAdapter(ind, (DishInterfaceClass) this);
                 binding.recview.setAdapter(adapter);
                 break;


             case 8:
                 FirebaseRecyclerOptions<DishesClass> tom=
                     new FirebaseRecyclerOptions.Builder<DishesClass>()
                                 .setQuery(
 FirebaseDatabase.getInstance().getReference().child("Koshry"), DishesClass.class)
                                 .build();
                 adapter = new DishesAdapter(tom, (DishInterfaceClass) this);
                 binding.recview.setAdapter(adapter);
                 break;

             case 9:
                 FirebaseRecyclerOptions<DishesClass> fsh=
                         new FirebaseRecyclerOptions.Builder<DishesClass>()
                                 .setQuery(FirebaseDatabase.getInstance().getReference().child("fish"), DishesClass.class)
                                 .build();
                 adapter = new DishesAdapter(fsh,(DishInterfaceClass) this);
                 binding.recview.setAdapter(adapter);
                 break;

         }

     }

     @Override
     protected void onStart() {
         super.onStart();
         adapter.startListening();
     }

     @Override
     protected void onStop() {
         super.onStop();
         adapter.stopListening();
     }

     @Override
     public void onItemClick(int position) {
         Toast.makeText(Dishes.this,"Added to your
 cart",Toast.LENGTH_LONG).show();

     }

     }

```

```java
package com.example.WaGbA;

public class Orders {

    private String Mealdetails,OrdeSstatus,Quantity;

    public Orders(){}

    public Orders(String mealdetails, String
ordeSstatus, String quantity) {
        Mealdetails = mealdetails;
        OrdeSstatus = ordeSstatus;
        Quantity = quantity;
    }

    public String getMealdetails() {
        return Mealdetails;
    }

    public void setMealdetails(String mealdetails) {
        Mealdetails = mealdetails;
    }

    public String getOrdeSstatus() {
        return OrdeSstatus;
    }

    public void setOrdeSstatus(String ordeSstatus) {
        OrdeSstatus = ordeSstatus;
    }

    public String getQuantity() {
        return Quantity;
    }

    public void setQuantity(String quantity) {
        Quantity = quantity;
    }
}

```

```java
package com.example.WaGbA;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.util.Patterns;
import android.view.View;
import android.widget.Toast;

import com.example.WaGbA.databinding.ActivitySignInBinding;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;

import java.util.regex.Pattern;

public class SignIn extends AppCompatActivity {

    private ActivitySignInBinding binding;
    private FirebaseAuth mAuth;
    private Intent mainPage;
    public static final Pattern School_Email
            = Pattern.compile(
            "[a-zA-Z0-9\\+\\.\\_\\%\\-\\+]{1,256}" +
                    "\\@" +"eng.asu.edu.eg"
    );

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding = ActivitySignInBinding.inflate(getLayoutInflater());
        View view = binding.getRoot();
        setContentView(view);


        mAuth = FirebaseAuth.getInstance();
        mainPage=new Intent(this, RestaurantsPage.class);
        binding.createbtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                createAccount();

            }
        });

    }

    private void createAccount() {

        String name=binding.nameEdt.getText().toString();
        String mail=binding.mailEdt.getText().toString();
        String phone=binding.phoneEdt.getText().toString();
        String address=binding.addrEdt.getText().toString();
        String password=binding.passEdt.getText().toString();

        if(name.isEmpty()){
            binding.nameEdt.setError("Name is required!");
            binding.nameEdt.requestFocus();
            return;
        }
        if(mail.isEmpty()){
            binding.mailEdt.setError("Email is required!");
            binding.mailEdt.requestFocus();
            return;
        }

        if(!Patterns.PHONE.matcher(phone).matches()){
            binding.phoneEdt.setError("Please provide a valid phone number!");
            binding.phoneEdt.requestFocus();
            return;

        }
        if(address.isEmpty()){
            binding.addrEdt.setError("Address is required!");
            binding.addrEdt.requestFocus();
            return;
        }
        if(password.isEmpty()){
            binding.passEdt.setError("Password is required!");
            binding.passEdt.requestFocus();
            return;
        }
        if(password.length()<6){
            binding.passEdt.setError("Password length must be >=6 characters!");
            binding.passEdt.requestFocus();
            return;
        }

        if(!School_Email.matcher(mail).matches()){
            binding.mailEdt.setError("Please provide a valid email address!");
            binding.mailEdt.requestFocus();
            return;

        }

        UserEntity userEntity=new UserEntity();
        userEntity.setUsername(binding.nameEdt.getText().toString());
        userEntity.setUsermail(binding.mailEdt.getText().toString());
```

```java
        userEntity.setUserphone(Integer.parseInt(binding.phoneEdt.getText().toString()));
        userEntity.setUseraddress(binding.addrEdt.getText().toString());
        userEntity.setUserpassword(binding.passEdt.getText().toString());

        UserDatabase userDatabase=UserDatabase.getUserDatabase(getApplicationContext());
        UserDao userDao=userDatabase.userDao();
        runOnUiThread(new Runnable() {
            @Override
            public void run() {

            }
        });

            mAuth.createUserWithEmailAndPassword(mail, password)
                    .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                        @Override
                        public void onComplete(@NonNull Task<AuthResult> task) {
                            if (task.isSuccessful()) {
                                Toast.makeText(SignIn.this, "Successfully Signed In!", Toast.LENGTH_LONG).show();
                                startActivity(mainPage);
                                finish();
                            } else {
                                // If sign in fails, display a message to the user.
                                Toast.makeText(SignIn.this, "Sign in failed!Make another trial!", Toast.LENGTH_LONG).show();

                            }
                        }
                    });
        }
    }


```

```java
package com.example.WaGbA;

import androidx.room.Dao;
import androidx.room.Insert;

@Dao
public interface UserDao {
    @Insert
    void registerUser(UserEntity userEntity);
}

```

```java
package com.example.WaGbA;

import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.Display;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.bumptech.glide.Glide;
import com.cepheuen.elegantnumberbutton.view.ElegantNumberButton;
import com.firebase.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

import java.util.HashMap;

public class CartPage extends AppCompatActivity {

    private TextView totPrice, totAmount, Delivery, Display,availability;
    private EditText userID;
    private ImageView img;
    private Intent receivePrice,receiveImg;
    private Button confirm,remove,add,sub,cash,credit;
    private int p,totalQuantity=1;
    private FirebaseDatabase firebaseDatabase;
    private DatabaseReference databaseReference;
    private HashMap<String ,Object> orders;
    private String pMethod;






    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_cart_page);

        firebaseDatabase=FirebaseDatabase.getInstance();
        databaseReference=firebaseDatabase.getReference().child("Orders").child("Order Details");


        img=findViewById(R.id.img1);
        totPrice=findViewById(R.id.tPrice);
        totAmount=findViewById(R.id.total);
        Delivery=findViewById(R.id.delivery);
        add=findViewById(R.id.addbtn);
        sub=findViewById(R.id.subbtn);
        userID=findViewById(R.id.userId);
        availability=findViewById(R.id.availTv);
        cash=findViewById(R.id.cash);
        credit=findViewById(R.id.credit);

        Display=findViewById(R.id.displayedt);
        receivePrice=getIntent();
        receiveImg=getIntent();
        String url=receiveImg.getStringExtra("orderImg");
        Glide.with(this).load(url).into(img);
        String price = receivePrice.getStringExtra("price");

        totPrice.setText(price);
        p= Integer.parseInt(totPrice.getText().toString());
        totAmount.setText(String.valueOf(p*totalQuantity));

        cash.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                pMethod="Cash";

            }
        });
        credit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
               pMethod="Credit";
            }
        });

      add.setOnClickListener(new View.OnClickListener() {
          @Override
          public void onClick(View v) {
              if(totalQuantity<100) {
                  totalQuantity++;
                  Display.setText(String.valueOf(totalQuantity));
                  totAmount.setText(String.valueOf(p*totalQuantity));
                  if(totalQuantity>=10){
                      availability.setText("unavailable");
                  }

              }
          }
      });

      sub.setOnClickListener(new View.
```

```java
OnClickListener() {
          @Override
          public void onClick(View v) {
              if (totalQuantity > 1) {
                  totalQuantity--;
                  Display.setText(String.valueOf(totalQuantity));
                  totAmount.setText(String.valueOf(p * totalQuantity));
                  if(totalQuantity<=10){
                      availability.setText("available");
                  }
              }
          }
      });



        confirm=findViewById(R.id.confirm);
        confirm.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v){

                if(availability.getText().toString()=="unavailable"){
                    availability.setError("we have only 10 dishes of this meal!");
                    availability.requestFocus();
                    return;

                }else {

                    Intent OrderStatIntent = new Intent(CartPage.this, OrderStatus.class);
                    orders = new HashMap<>();
                    orders.put("meal details",url);
                    orders.put("Quentity", totalQuantity);
                    orders.put("payment method",
pMethod);
                    String msg = "Waiting for acceptance";
                    orders.put("Order status", msg);
                    if (userID.getText().toString().isEmpty()) {
                        userID.setError("your id is required to track your order!");
                        userID.requestFocus();
                        return;

                    } else if (userID.length() < 7 ) {
                        userID.setError("your id must be<=7!");
                        userID.requestFocus();
                        return;

                    } else {
                        orders.put("UserID", userID.getText().toString());
                    }
                    databaseReference.setValue(orders);
                    startActivity(OrderStatIntent);
                }
            }
        });

        remove=findViewById(R.id.Remove);
        remove.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                databaseReference.setValue(null);
                Intent intent=new Intent(CartPage.this,Dishes.class);
                startActivity(intent);

            }
            });



        }
}
```

```java
package com.example.WaGbA;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;

import com.example.WaGbA.databinding.ActivityHomePageBinding;

public class HomePage extends AppCompatActivity {

    private ActivityHomePageBinding binding;
    Intent loginpage,signpage;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding = ActivityHomePageBinding.inflate(getLayoutInflater());
        View view = binding.getRoot();
        setContentView(view);
        loginpage=new Intent(this,LoginPage.class);
        signpage=new Intent(this,SignIn.class);


        binding.logbtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                startActivity(loginpage);

            }
        });
        binding.signbtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                startActivity(signpage);

            }
        });
    }
}
```

```java
package com.example.WaGbA;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Toast;

import com.example.WaGbA.databinding.ActivityLoginPageBinding;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;

public class LoginPage extends AppCompatActivity {
    private FirebaseAuth mAuth;
    private ActivityLoginPageBinding binding;
    Intent mainPage;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding = ActivityLoginPageBinding.inflate(getLayoutInflater());
        View view = binding.getRoot();
        setContentView(view);
        mAuth = FirebaseAuth.getInstance();
        mainPage=new Intent(this, RestaurantsPage.class);

        binding.loginbtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                binding.loginbtn.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View view) {
                        loginAccount();
                    }
                });

            }
        });
    }

    private void loginAccount() {

        String mail = binding.mailtxt.getText().toString();
        String password = binding.passwordtxt.getText().toString();

        if (mail.isEmpty()) {
            binding.mailtxt.setError("Email is required!");
            binding.mailtxt.requestFocus();
            return;
        }
        if (password.isEmpty()) {
            binding.passwordtxt.setError("Password is required!");
            binding.passwordtxt.requestFocus();
            return;
        }

        mAuth.signInWithEmailAndPassword(mail, password)
                .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                    @Override
                    public void onComplete(@NonNull Task<AuthResult> task) {
                        if (task.isSuccessful()) {
                            Toast.makeText(LoginPage.this, "Logged in successfully!", Toast.LENGTH_LONG).show();
                            startActivity(mainPage);
                            finish();


                        } else {
                            // If sign in fails, display a message to the user.
                            Toast.makeText(LoginPage.this,
                                    "Email or Password is incorrect please try again", Toast.LENGTH_LONG).show();

                        }
                    }
                });



    }
}
```

```java
package com.example.WaGbA;

import androidx.room.ColumnInfo;
import androidx.room.Entity;
import androidx.room.PrimaryKey;

@Entity(tableName = "user")
public class UserEntity {
    @PrimaryKey(autoGenerate = true)
    int id;

    @ColumnInfo(name = "username")
    String username;

    @ColumnInfo(name = "usermail")
    String usermail;

    @ColumnInfo(name = "userPhone")
    int userphone;

    @ColumnInfo(name = "useraddress")
    String useraddress;

    @ColumnInfo(name = "userpassword")
    String userpassword;

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public String getUsermail() {
        return usermail;
    }

    public void setUsermail(String usermail) {
        this.usermail = usermail;
    }

    public int getUserphone() {
        return userphone;
    }

    public void setUserphone(int userphone) {
        this.userphone = userphone;
    }

    public String getUseraddress() {
        return useraddress;
    }

    public void setUseraddress(String useraddress) {
        this.useraddress = useraddress;
    }

    public String getUserpassword() {
        return userpassword;
    }

    public void setUserpassword(String userpassword) {
        this.userpassword = userpassword;
    }
}

```

```java
package com.example.WaGbA;

public class DishesClass {

    String purl,price;
    DishesClass()
    {

    }

    public DishesClass(String purl, String price) {
        this.purl = purl;
        this.price = price;
    }

    public String getPrice() {
        return price;
    }

    public void setPrice(String price) {
        this.price = price;
    }

    public String getPurl() {
        return purl;
    }

    public void setPurl(String purl) {
        this.purl = purl;
    }
}
```

```java
package com.example.WaGbA;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

public class OrderStatus extends AppCompatActivity {
    private TextView accept;
    private FirebaseDatabase firebaseD;
    private DatabaseReference readData;
    private EditText UsrMail;
    private Button lsbtn,Restaurant;


    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_order_status);

        UsrMail=findViewById(R.id.usrml);
        firebaseD=FirebaseDatabase.getInstance();
        accept=findViewById(R.id.confTv);
         lsbtn=findViewById(R.id.lst);
         Restaurant=findViewById(R.id.button);
         Restaurant.setOnClickListener(new View.OnClickListener() {
             @Override
             public void onClick(View v) {
                 Intent intent=new Intent(OrderStatus.this,RestaurantsPage.class);
                 startActivity(intent);
             }
         });
         firebaseD=FirebaseDatabase.getInstance();
         lsbtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {



                if(!UsrMail.getText().toString().isEmpty()) {

                    readData=firebaseD.getReference().child("Accepted Orders")
                            .child(UsrMail.getText().toString());

                    readData.addListenerForSingleValueEvent(new ValueEventListener() {
                        @Override
                        public void onDataChange(@NonNull DataSnapshot snapshot) {
                            String val = snapshot.getValue().toString();
                            accept.setText(val);
                        }

                        @Override
                        public void onCancelled(@NonNull DatabaseError error) {

                        }
                    });
                }else { accept.setText(" ");

            }
            }
        });

    }

}
```

```java
package com.example.WaGbA;

import android.content.Context;

import androidx.room.Database;
import androidx.room.Room;
import androidx.room.RoomDatabase;

@Database(entities = {UserEntity.class},version = 1)

public abstract class UserDatabase extends RoomDatabase {
    private static final String dbName="user";
    private static UserDatabase userDatabase;
    public static synchronized UserDatabase getUserDatabase(Context context){

        if(userDatabase==null){
            userDatabase= Room.databaseBuilder(context,UserDatabase.class,dbName).fallbackToDestructiveMigration().build();
        }
        return userDatabase;
    }

    public abstract UserDao userDao();
}

```

```java
 package com.example.WaGbA;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import android.content.Intent;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import com.bumptech.glide.Glide;
import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.firebase.ui.database.FirebaseRecyclerOptions;

 public class DishesAdapter extends FirebaseRecyclerAdapter<DishesClass,DishesAdapter.myviewholder> {

    private final DishInterfaceClass dishInterfaceClass;

    public DishesAdapter(@NonNull FirebaseRecyclerOptions<DishesClass> options, DishInterfaceClass dishInterfaceClass) {
        super(options);
        this.dishInterfaceClass = dishInterfaceClass;
    }

    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position, @NonNull DishesClass DishesClass) {
        holder.pricetxt.setText(DishesClass.getPrice());
        Glide.with(holder.img.getContext()).load(DishesClass.getPurl()).into(holder.img);
        holder.itemView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent=new Intent(holder.img.getContext(),CartPage.class);
                intent.putExtra("price",DishesClass.getPrice());
                intent.putExtra("orderImg",DishesClass.getPurl());
                holder.img.getContext().startActivity(intent);

            }
        });
    }

    @NonNull
    @Override
    public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view= LayoutInflater.from(parent.getContext()).inflate(R.layout.dishes_item,parent,false);
        return new myviewholder(view,dishInterfaceClass);
    }

    class myviewholder extends RecyclerView.ViewHolder {
       ImageView img;
       TextView pricetxt;
       public myviewholder(@NonNull View itemView, DishInterfaceClass dishInterfaceClass)
       {
           super(itemView);
           img=itemView.findViewById(R.id.img1);
           pricetxt=(TextView)itemView.findViewById(R.id.priceTxt);

           itemView.setOnClickListener(new View.
OnClickListener() {
                @Override
                public void onClick(View v) {
                    if( DishesAdapter.this.dishInterfaceClass !=null){
                        int pos= getAdapterPosition();

                        if(pos!=RecyclerView.NO_POSITION){
                            DishesAdapter.this.dishInterfaceClass.onItemClick(pos);
                        }
                    }
                }
            });

        }
    }

}


```

```java
package com.example.WaGbA;

import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;


import com.example.WaGbA.databinding.ActivityRestaurantsPageBinding;
import com.firebase.ui.database.FirebaseRecyclerOptions;
import com.google.firebase.database.FirebaseDatabase;

public class RestaurantsPage extends AppCompatActivity implements DishInterfaceClass
{
    private ActivityRestaurantsPageBinding binding;
    private RestaurantsAdapter adapter;
    private  Intent orderIntent,CartIntent;
    private Button Order;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding = ActivityRestaurantsPageBinding .inflate(getLayoutInflater());
        View view = binding.getRoot();
        setContentView(view);
        binding.recview.setLayoutManager(new LinearLayoutManager(this));
        Order=findViewById(R.id.orderPagebtn);
        Order.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent=new Intent(RestaurantsPage.this,OrderStatus.class);
                startActivity(intent);
            }
        });



        FirebaseRecyclerOptions<RestaurantsClass> options =
                new FirebaseRecyclerOptions.Builder<RestaurantsClass>()
                        .setQuery(FirebaseDatabase.getInstance().getReference().child("Restaurants"), RestaurantsClass.class)
                        .build();

        adapter=new RestaurantsAdapter(options, this);
        binding.recview.setAdapter(adapter);

    }

    @Override
    protected void onStart() {
        super.onStart();
        adapter.startListening();
    }

    @Override
    protected void onStop() {
        super.onStop();
        adapter.stopListening();
    }

    @Override
    public void onItemClick(int position) {
        Intent intent=new Intent(RestaurantsPage.this,Dishes.class);
        intent.putExtra("pos",position);
        startActivity(intent);
        }
    }

```

```java
package com.example.WaGbA;

public class RestaurantsClass
{
  String name,purl;
    RestaurantsClass()
    {

    }
    public RestaurantsClass(String name, String purl) {
        this.name = name;
        this.purl = purl;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }


    public String getPurl() {
        return purl;
    }

    public void setPurl(String purl) {
        this.purl = purl;
    }
}

```

```java
package com.example.WaGbA;

public interface DishInterfaceClass {
    void onItemClick(int position );
}

```

```java
package com.example.WaGbA;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import com.bumptech.glide.Glide;
import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.firebase.ui.database.FirebaseRecyclerOptions;

public class RestaurantsAdapter extends FirebaseRecyclerAdapter<RestaurantsClass, RestaurantsAdapter.myviewholder>
{
    private final DishInterfaceClass dishInterfaceClass;

    public RestaurantsAdapter(@NonNull FirebaseRecyclerOptions<RestaurantsClass> options, DishInterfaceClass dishInterfaceClass) {
        super(options);
        this.dishInterfaceClass = dishInterfaceClass;
    }

    @Override
    protected void onBindViewHolder(@NonNull myviewholder holder, int position, @NonNull RestaurantsClass RestaurantsClass)
    {
       holder.name.setText(RestaurantsClass.getName());
       Glide.with(holder.img.getContext()).load(RestaurantsClass.getPurl()).into(holder.img);
    }

    @NonNull
    @Override
    public myviewholder onCreateViewHolder(@NonNull ViewGroup parent, int viewType)
    {
       View view= LayoutInflater.from(parent.getContext()).inflate(R.layout.restaurants_iteem, parent,false);
       return new myviewholder(view, dishInterfaceClass);
    }

    class myviewholder extends RecyclerView.ViewHolder
    {
        ImageView img;
        TextView name;
        public myviewholder(@NonNull View itemView, DishInterfaceClass dishInterfaceClass)
        {
            super(itemView);
            img=itemView.findViewById(R.id.img1);
            name=(TextView)itemView.findViewById(R.id.nametext);

            itemView.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {
                    if( RestaurantsAdapter.this.dishInterfaceClass !=null){
                        int pos= getAdapterPosition();

                        if(pos!=RecyclerView.NO_POSITION){
                            RestaurantsAdapter.this.dishInterfaceClass.onItemClick(pos);
                        }
                    }
                }
            });

        }
    }
}

```