Question 1:

Suppose you have the following event handler:

public void clickHandler(View v){

//logic

}

Why is the datatype of the input parameter declared as ‘View’?

View is a base class for widgets, which are used to create interactive UI components (button, text fields, etc.) So by declaring the input parameter, upcasting is applied and that method (event handler) could be used for multiple UI component instances such as Button, text field, etc. without needing to specifying which View objects are being accessed.

Question 2:

1. What is the main difference between Guidelines and Barriers?

Barrier’s position is flexible and always based on the size of multiple UI elements contained within it, whereas Guideline’s position is always fixed

1. When does the horizontal bias in constraint become redundant?

When the views are in a horizontal chain, giving bias to one of the view will also automatically change the bias of the connected views since they are bidirectionally connected to each other.

Question 3:

1. What is the role of the second input parameter to BroadcastReceiver onRecieve() method?

The second input parameter is used to specify the messaging object that can be used to request an action from another app component. So the action can be identified and carried out.

1. If you have the following:
   1. Intent Filter =“week4.fit2081.quiz”
   2. Key = “Key”
   3. Data=“FIT2081”

Using the details above, write a piece of code that sends the Data in a broadcast message

Void sendData (Context context){

Intent msgIntent = new Intent();

msgIntent.setAction(“week4.fit2081.quiz”);

msgIntent.putExtra(“Key”,”FIT2081”);

context.sendBroadcast(msgIntent);

}

Question 4:

public class MainActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);

MyBroadCastReceiver myBroadCastReceiver = new MyBroadCastReceiver();

registerReceiver(myBroadCastReceiver, new IntentFilter("QuizFilter")

}