**TEST ANSWER SHEET**

**Name : Enggar N.E. Putra**

#### Basic Programming

1. Integer
2. Char
3. Result will be ”**113**\r\n”
4. Result will be “**0**\r\n**0**\r\n**1**\r\n**1**\r\n**1**\r\n**2**\r\n”
5. Result will be “**3566GoldenStreetMiami**\r\n”

#### Flaky Test

1. In general, it is a test of an application/program which succeed in one time but fail in another time without any change in the application/program/code. This means the application/program not yet ready to deliver since it is not guaranteed to work.
2. Set a series of data as input and a series of expected-result. Compare the expected-result with the real-result. Repeat the step several times as required to satisfy the requirement that real-result has matched the expected result.
3. Ask developer to create a back-door which is special for test mode.

#### Essay Question Topic

1a.

**package** test\_package;

**import** java.io.BufferedReader;

**import** java.io.IOException;

**import** java.io.InputStreamReader;

**import** java.net.HttpURLConnection;

**import** java.net.URL;

**import** org.json.JSONArray;

**import** org.json.JSONObject;

**public** **class** MyClass1 {

**public** **static** **void** main(String[] args) **throws** IOException {

// my variables

String myText = "";

// **TODO** Auto-generated method stub

// Create a neat value object to hold the URL

URL url = **new** URL("https://jsonplaceholder.typicode.com/posts");

// Open a connection(?) on the URL(?) and cast the response(??)

HttpURLConnection connection = (HttpURLConnection) url.openConnection();

// Now it's "open", we can set the request method, headers etc.

connection.setRequestProperty("accept", "application/json");

connection.setRequestMethod("GET");

**int** status = connection.getResponseCode();

System.***out***.println(status);

BufferedReader in = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream()));

String inputLine;

StringBuffer content = **new** StringBuffer();

**while** ((inputLine = in.readLine()) != **null**) {

content.append(inputLine);

}

in.close();

myText = content.toString();

JSONArray array = **new** JSONArray(myText);

JSONObject object = **new** JSONObject();

**boolean** errorPresent = **false**;

System.***out***.printf("Checking %d data\r\n", array.length());

**for**(**int** i=0; i < array.length(); i++) {

object = array.getJSONObject(i);

**try**{

object.getInt("userId");

object.getInt("id");

object.getString("title");

object.getString("body");

}

**catch**(Exception e){

System.***out***.println(e);

System.***out***.printf("Data number %d is bad\r\n", i);

errorPresent = **true**;

}

}

**if**(!errorPresent) {

System.***out***.printf("%d data without error\r\n", array.length());

}

System.***out***.println("The 1st element of array");

System.***out***.println(array.get(0));

System.***out***.println("The 100th element of array");

System.***out***.println(array.get(99));

connection.disconnect();

}

}

1b.

**package** test\_package;

**import** java.io.BufferedReader;

**import** java.io.DataOutputStream;

**import** java.io.IOException;

**import** java.io.InputStreamReader;

**import** java.net.HttpURLConnection;

**import** java.net.URL;

**import** org.json.JSONArray;

**import** org.json.JSONObject;

**public** **class** MyClass2 {

**public** **static** **void** main(String[] args) **throws** IOException {

// my variables

JSONObject data = **new** JSONObject();

data.put("title", "recommendation");

data.put("body", "motorcycle");

data.put("id", 12);

String myText = data.toString();

// **TODO** Auto-generated method stub

// Create a neat value object to hold the URL

URL url = **new** URL("https://jsonplaceholder.typicode.com/posts");

// Open a connection(?) on the URL(?) and cast the response(??)

HttpURLConnection connection = (HttpURLConnection) url.openConnection();

// Now it's "open", we can set the request method, headers etc.

connection.setRequestMethod("POST");

connection.setRequestProperty("accept", "application/json");

connection.setRequestProperty("content-type", "application/json; utf-8");

connection.setDoOutput(**true**);

DataOutputStream wr = **new** DataOutputStream( connection.getOutputStream());

wr.writeUTF(myText);

BufferedReader in = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream()));

String inputLine;

StringBuffer content = **new** StringBuffer();

**while** ((inputLine = in.readLine()) != **null**) {

content.append(inputLine);

}

in.close();

**int** status = connection.getResponseCode();

System.***out***.println(status);

JSONArray array = **new** JSONArray();

JSONObject object = **new** JSONObject();

**if**( status != 201) {

System.***out***.println("Failed to POST!");

}**else** {

System.***out***.println(connection.getContentType());

System.***out***.println(content);

myText = content.toString();

**try** {

**try** {

array = **new** JSONArray(myText);

object = array.getJSONObject(100);

}**catch**(Exception e) {

System.***out***.println(e);

**try** {

object = **new** JSONObject(myText);

}**catch**(Exception e2) {

System.***out***.println(e2);

System.***out***.println("is not a JSON, exiting...");

System.*exit*(-1);

}

}

**if**(object.has("id") && object.has("userId") && object.has("title") && object.has("body")) {

**if**(object.get("id").equals(101) && object.get("userId").equals(12) && object.get("title").toString().equals("recommendation") && object.get("body").toString().equals("motorcycle")){

System.***out***.println("Input == Output, POST succeeded");

connection.disconnect();

System.*exit*(0);

}**else**{

System.***out***.println("Input != Output, POST failed");

connection.disconnect();

System.*exit*(-1);

}

}**else** {

System.***out***.println("response is incomplete");

}

}**catch**(Exception e){

System.***out***.println(e);

}

}

connection.disconnect();

// This line makes the request

// Open a connection(?) on the URL(?) and cast the response(??)

connection = (HttpURLConnection) url.openConnection();

// Now it's "open", we can set the request method, headers etc.

connection.setRequestMethod("GET");

connection.setRequestProperty("accept", "application/json");

in = **new** BufferedReader(**new** InputStreamReader(connection.getInputStream()));

inputLine = "";

content.delete(0, content.length());

**while** ((inputLine = in.readLine()) != **null**) {

content.append(inputLine);

}

in.close();

status = connection.getResponseCode();

System.***out***.println(status);

**if**( status != 200) {

System.***out***.println("Failed to GET!");

}**else** {

System.***out***.println(connection.getContentType());

System.***out***.println(content);

myText = content.toString();

**try** {

**try** {

array = **new** JSONArray(myText);

object = array.getJSONObject(100);

}**catch**(Exception e) {

System.***out***.println(e);

**try** {

object = **new** JSONObject(myText);

}**catch**(Exception e2) {

System.***out***.println(e2);

System.***out***.println("is not a JSON, exiting...");

System.*exit*(-1);

}

}

**if**(object.has("id") && object.has("userId") && object.has("title") && object.has("body")) {

**if**(object.get("id").equals(101) && object.get("userId").equals(12) && object.get("title").toString().equals("recommendation") && object.get("body").toString().equals("motorcycle")){

System.***out***.println("Input == Output, POST succeeded");

connection.disconnect();

System.*exit*(0);

}**else**{

System.***out***.println("Input != Output, POST failed");

connection.disconnect();

System.*exit*(-1);

}

}**else** {

System.***out***.println("response is incomplete");

}

}**catch**(Exception e){

System.***out***.println(e);

}

}

connection.disconnect();

}

}

2. Not yet completed

3. Not yet completed