**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **12/06/2020** | **Name:** | **PRIYA P RAO** |
| **Course:** | **PCB Design** | **USN:** | **4AL18EC041** |
| **Topic:** | * **Design principles and basic concepts.** * **Additional design considerations.** | **Semester & Section:** | **4TH sem ‘A’ Section.** |
| **Github Repository:** | **Priya-Rao** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session**  **C:\Users\Pawan\Desktop\gryht.PNG**  **C:\Users\Pawan\Desktop\tfjhj.PNG** |
| **In today’s session I have studied about:**  **Chapter 1: Design principles and basic concepts.**  **The schematic design process**  **Given below are the 8 steps of schematic design workflow:** SetupSymbolsPlace and annotate symbolsWiringNetsElectrical Rules CheckCommentsNetlistChapter 2: Additional design considerations.In this chapter I have studied about:  * [**Design Rules**](https://www.botfactory.co/blog/tutorials-and-demos-2/post/kicad-design-guide-92#design_rules) * [**Create an Outline**](https://www.botfactory.co/blog/tutorials-and-demos-2/post/kicad-design-guide-92#set_outline) * [**Set an Origin**](https://www.botfactory.co/blog/tutorials-and-demos-2/post/kicad-design-guide-92#set_origin) * [**Gerber File Creation**](https://www.botfactory.co/blog/tutorials-and-demos-2/post/kicad-design-guide-92#gerber_file_creation) * [**Drill File Creation**](https://www.botfactory.co/blog/tutorials-and-demos-2/post/kicad-design-guide-92#drill_file_creation) * [**Centre and Rotation File Creation**](https://www.botfactory.co/blog/tutorials-and-demos-2/post/kicad-design-guide-92#position_file_creation) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **12/06/2020** | **Name:** | **PRIYA P RAO** |
| **Course:** | **JAVA** | **USN:** | **4AL18EC041** |
| **Semester & Section:** | **4th sem ‘A’ Section.** | | |
| **Github Repository:** | **Priya-Rao** |  |  |

|  |
| --- |
| **AFTERNOON SESSION DETAILS** |
| **Image of session**  **C:\Users\Pawan\Desktop\qswsaxwd.PNG**  **C:\Users\Pawan\Desktop\jgujgyjy.PNG** |
| **In today’s session I have learnt about:**   * **Generics and Wildcards: Wildcards in Java. The question mark (?) is known as the wildcard in generic programming . It represents an unknown type.** * **Anonymous Classes: It is an inner class without a name and for which only a single object is created.** * **Reading Files Using Scanner:**   **import java.io.File;**  **import java.io.IOException;**  **import java.util.Scanner;**  **class Main**  **{**  **public static void main(String[] args) {**  **File file = new File("doc.txt");**  **try (Scanner sc = new Scanner(file, StandardCharsets.UTF\_8.name())) {**  **while (sc.hasNextLine()){**  **System.out.println(sc.nextLine());**  **}**  **}**  **catch (IOException e) {**  **e.printStackTrace();**  **}**  **}**  **}**   * **Handling Exceptions: The Exception Handling in Java is one of the powerful mechanism to handle the runtime errors so that normal flow of the application can be maintained.** * **Multiple Exceptions: In Java 7 it was made possible to catch multiple different exceptions in the same catch block. This is also known as multi catch.** * **Runtime vs. Checked Exceptions: 2) Unchecked are the exceptions that are not checked at compiled time.** * **Abstract Classes: A class which is declared with the abstract keyword is known as an abstract class in**[**Java**](https://www.javatpoint.com/java-tutorial)**.** * **Reading Files With File Reader: There are many ways to read a file to String in Java. We will explore the following ways:** * **Java read file to String using [BufferedReader](https://www.journaldev.com/19879/java-bufferedreader" \t "_blank)** * **Read file to String in java using [FileInputStream](https://www.journaldev.com/19187/java-fileinputstream" \t "_blank)** * **Java read file to string using**[**Files class**](https://www.journaldev.com/17794/java-files-nio-files-class) * **Read file to String using**[**Scanner**](https://www.journaldev.com/872/scanner-class-in-java)**class** * **Java read file to string using Apache Commons IO FileUtils class** * **Try-With-Resources: The try-with-resources statement is a try statement that declares one or more resources. A resource is an object that must be closed after the program is finished with it. The try-with-resources statement ensures that each resource is closed at the end of the statement. Any object that implements java.lang.AutoCloseable, which includes all objects which implement java.io.Closeable, can be used as a resource** * **Creating and Writing Text Files: To create a file in Java, you can use the createNewFile() method. This method returns a boolean value: true if the file was successfully created, and false if the file already exists. Note that the method is enclosed in a try...catch block. This is necessary because it throws an IOException if an error occurs (if the file cannot be created for some reason):** |