|  |
| --- |
| Hands-on Exercise Objective |
| After completing the hands-on exercises, you will be able to:   * Declare constructors * Overload constructors * Chain the constructor calling * Apply access specifiers to constructors |
| **Problem Statement 1: Declaring and using constructors**  Create a class ***Circle.java*** in a package “***com.capgemini.shapes***” , add a float instance variable ***radius*** and add a default constructor (**Constructor 1**) for the class. This constructor should initialize the radius to a default value ***1.5f***.  The above constructor should be invoked from a main method from another class, ***Shape.java*** (in different package ***com.capgemini. geometry***).  **Problem Statement 2: Overloading constructors and using “this” keyword.**  In the Circle.java class created above add an instance float variable ***pi*** and create two overloaded constructors.  **Constructor 2**- with a float argument name ***radius.*** The constructor should initialize the class variable ***radius***  with the method argument radius.  **NOTE:**  The instance variable and the method argument should be named same as “***radius***”.  **Constructor 3**- with two float arguments ***radius and pi***. Default the class pi value to 3.5 and set the instance variable with the radius method argument.  The constructor “***constructor 2”*** should be invoked from a main method from class, ***Area.java*** (in a package ***com.capgemini.shapes***).  **Problem Statement 3: Constructor Chaining**  In Circle.java, invoke the Constructor 3 created in the previous step from Constructor 2.  **Problem Statement 4: Applying access specifiers to constructors/variables**   1. (Other classes must not be able to call this constructor). Also restrict the access to the variable radius to class level 2. Provide package level access to Constructor 2 (Classes in other package must **not** be able to access this constructor). Also provide package level access to the variable pi.   **Problem Statement 5:** Create two methods and calculate area and circumference of a Circle  In the **Circle.java** class, create two methods as listed below   1. Method 1 - calculateCircleArea should accept the float radius as parameter and calculate the area (pi\*r\*r). It should return the result value to the main method where it should be printed in the console. 2. Method 2 – calculateCircumference should accept float radius as parameter and calculate the circumference (2 \* pi \* r). It should return the result value to the main method where it should be printed in the console.   Call these two methods from the main method in **Circle.java** by passing appropriate parameters. |