**MODULE : 2**

**SECTION 2 :**

**Problem 1:**

**Simple ASCII Art  
Overview  
Using text to create a picture is known as ASCII art. In section 2, we made an ASCII art cat. In this practice, you’ll use print statements  
to recreate the image above.  
Task  
Use 8 print statements to recreate the smiley face above. Your art will rely on only a single character, besides space, such as X or #.  
The ProblemSet2\_1 project is available to help you get started.**

public class Main

{

public static void main(String[] args) {

String y=" ",x="⬜ ";

System.out.println(y+y+y+y+y+y+y+y);

System.out.println(y+y+x+x+x+x+y+y);

System.out.println(y+x+y+y+y+y+x+y);

System.out.println(x+y+x+y+y+x+y+x);

System.out.println(x+y+y+y+y+y+y+x);

System.out.println(x+y+x+y+y+x+y+x);

System.out.println(x+y+y+x+x+y+y+x);

System.out.println(y+x+y+y+y+y+x+y);

System.out.println(y+y+x+x+x+x+y+y);

}

}

**Problem 2:**

**Original ASCII Art  
Overview  
Using text to create a picture is known as ASCII art. In section 2, we made an ASCII art cat. In this practice, you’ll create your own  
beautiful work of art.  
Task  
Use print statements to create your own beautiful original ASCII art. Use comments to describe what your image is depicting.  
It’s ok for your art to rely on only a single character, besides space, such as X or #. But you’re encouraged to use a few different  
characters in your design, like in the cat example from class.  
Your art must also:  
• Use at least 8 print statements  
• Be at least 8 characters wide  
• Use at least 20 characters that aren’t space  
You’re welcome to create another cat. However, this image must look significantly different from the example used in class. Similarly,  
you’re welcome to create another face, but it must look significantly different from the face in the previous practice (it’s way too easy to  
turn the smiley face into a frowny face).  
Note: The backslash (\) character has special meaning in Java print statements. If you choose to use a backslash in your image, you’ll  
actually need to write two backslashes (\\) in your print statement.  
The ProblemSet2\_2 project is available to help you get started.**

public class Main

{

public static void main(String[] args) {

String y=" ";

System.out.println(y+y+y+y+y+y+y+y);

System.out.println(" /\\ /\\");

System.out.println(" / \\\_\_/ \\");

System.out.println(" / \\");

System.out.println(" ( /\\ /\\ )");

System.out.println(" ==== v ====");

System.out.println(" ======(\_|\_)=====");

System.out.println(" ( )");

System.out.println(" (\_\_\_\_)");

}

}

**Problem 3:**

**The Snake Box Factory  
Overview  
Dear Respectable Software Engineer,  
Here at the world renowned Snake Box Factory, we pride ourselves on our ability to deliver the highest quality, custom sized, cardboard  
boxes to our customers. Our boxes are filled with the highest quality, custom-ordered snakes. We service thousands of accounts  
worldwide and have a solid 98% satisfaction rating with customers. However, the entire ordering process is currently written on  
cardboard, which is transported between departments via carrier snake. We thought this would be a good way to show confidence in  
the quality and usefulness of our product. But as our business continues to grow, we’re realizing this was a bad idea. We believe it’s  
time for a more conventional and digitized approach to our operations. Would you be able to help us develop the software we need to  
make this happen?  
Sincerely,  
President George Johnson,  
The Snake Box Factory  
Tasks  
Read the scenario found in the overview and consider what objects could be modeled as part of creating a software solution. Identify 3  
objects from this scenario (remember, objects can be either tangible or abstract. List 3 properties and 3 behaviors belonging to each  
object.  
Write your solution as a document rather than a .java file.**

public class Box {

private int boxId;

private double length;

private double width;

private double height;

public Box(int boxId, double length, double width, double height) {

this.boxId = boxId;

this.length = length;

this.width = width;

this.height = height;

}

public void packBox() {

System.out.println("Box packed.");

}

}

public class Snake {

private int snakeId;

private double length;

public Snake(int snakeId, double length) {

this.snakeId = snakeId;

this.length = length;

}

public void transportSnake() {

System.out.println("Snake transported.");

}

}

public class Order {

private int orderId;

private Box box;

private Snake snake;

public Order(int orderId, Box box, Snake snake) {

this.orderId = orderId;

this.box = box;

this.snake = snake;

}

public void createOrder() {

System.out.println("Order created with ID: " + orderId);

}

}

public class Main {

public static void main(String[] args) {

Box box = new Box(1, 10.0, 5.0, 5.0);

Snake snake = new Snake(1, 3.0);

Order order = new Order(1, box, snake);

order.createOrder();

box.packBox();

snake.transportSnake();

    }

}