**section - 4.1**

**code:**

import java.util.Scanner;

public class Employee {

private final String name;

private final String username;

private final String email;

private String password;

// Constructor

public Employee() {

this.name = setName();

this.username = setUserName(name);

this.email = setEmail(username);

this.password = setPassword(username);

}

// Private method to count occurrences of a character in a string

private int countChars(String str, char ch) {

int count = 0;

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

if (str.charAt(i) == ch) {

count++;

}

}

return count;

}

// Method to get the employee's name from user input

private String setName() {

Scanner scanner = new Scanner(System.in);

String name;

while (true) {

System.out.print("Enter the employee's name (first and last name): ");

name = scanner.nextLine();

if (countChars(name, ' ') == 1) {

break;

} else {

System.out.println("Please enter both first and last name separated by a space.");

}

}

return name;

}

// Method to format the username

private String setUserName(String name) {

String[] parts = name.split(" ");

if (parts.length != 2) {

throw new IllegalArgumentException("Name should contain exactly first and last names.");

}

String firstName = parts[0].toLowerCase();

String lastName = parts[1].toLowerCase();

return firstName + "." + lastName;

}

// Method to format the email address

private String setEmail(String username) {

String[] parts = username.split("\\.");

if (parts.length != 2) {

throw new IllegalArgumentException("Username should be in the format 'firstname.lastname'");

}

String firstInitial = parts[0].substring(0, 1);

String lastName = parts[1];

return firstInitial + lastName + "@oracleacademy.Test";

}

// Method to generate the initial password

private String setPassword(String username) {

StringBuilder passwordBuilder = new StringBuilder(username.length());

for (char c : username.toCharArray()) {

if ("aeiou".indexOf(c) != -1) {

passwordBuilder.append('\*');

} else {

passwordBuilder.append(c);

}

}

// Ensure the password is exactly 8 characters long

while (passwordBuilder.length() < 8) {

passwordBuilder.append('\*');

}

if (passwordBuilder.length() > 8) {

passwordBuilder.setLength(8);

}

// Capitalize the first letter of the password

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

char ch = passwordBuilder.charAt(i);

if (Character.isLetter(ch)) {

passwordBuilder.setCharAt(i, Character.toUpperCase(ch));

break;

}

}

return passwordBuilder.toString();

}

@Override

public String toString() {

return String.format(

"Employee Details\nName : %s\nUsername : %s\nEmail : %s\nInitial Password : %s",

name, username, email, password

);

}

public static void main(String[] args) {

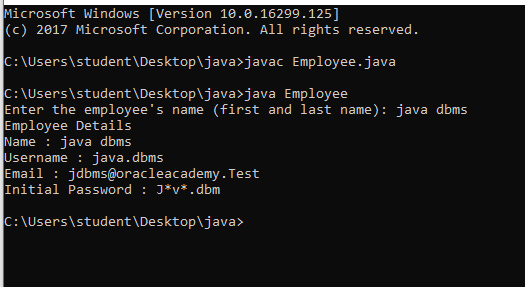
Employee employee = new Employee();

System.out.println(employee);

}

}

**output:**

****