**1. A company decided to give bonus of 5% to employee if his/her year of service is more than 5 years. Ask user for their salary and year of service and print the net bonus amount.**

**Sample Input: 20000, 7**

**Sample Output: 1000**

**Sample Input: 50000, 3**

**Sample Output: 0**

**Analysis:**

In this problem, we have to calculate the bonus amount based on the salary and year of service of the employee. If the year of service is more than 5 years, then the bonus amount will be 5% of the salary. Otherwise, the bonus amount will be 0. We will take the salary and year of service as input from the user and calculate the bonus amount based on the given conditions.

**Code :**

package com.self\_practice;

import java.util.**Scanner**;

public class **BonusSalary** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the salary amount: ");

int salary = sc.**nextInt**();

**System**.out.**print**("Enter your year of service: ");

int yearOfService = sc.**nextInt**();

int bonus = 0;

if (yearOfService > 5){

bonus = (int) (salary \* 0.05);

}

**System**.out.**println**("Bonus Amount: "+ bonus);

}

}

**Output :**

Enter the salary amount: 30000

Enter your year of service: 6

Bonus Amount: 1500

**2. Write a program to print absolute value of a number entered by user without using library method.**

**Sample Input: 1**

**Sample Output: 1**

**Sample Input: -1**

**Sample Output: 1**

**Analysis:**

In this problem, we have to calculate the absolute value of a number entered by the user without using any library method. The absolute value of a number is the positive value of the number, regardless of its sign. We will take the number as input from the user and calculate the absolute value of the number using the following conditions:

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **AbsoluteNumber** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter any number: ");

int userInput = sc.**nextInt**();

if (userInput < 0){

userInput \*= -1;

}

**System**.out.**println**("Absolute number of the given number: "+ userInput);

}

}

**Output:**

Enter any number: -5

Absolute number of the given number: 5

**3. There is a group selection in a crowd for an event based on the entry ticket number. For every group, the group leader will be the one who’s ticket number has 10 as it’s factor. If anyone gives the ticket number tell him/her that they are a group leader or a group member.**

**Sample Input: 5423**

**Sample Output: Group Member**

**Sample Input: 5610**

**Sample Output: Group Leader**

**Analysis:**

In this problem, we have to determine whether a person is a group leader or a group member based on their ticket number. If the ticket number is divisible by 10, then the person is a group leader. Otherwise, the person is a group member. We will take the ticket number as input from the user and check whether it is divisible by 10 to determine whether the person is a group leader or a group member.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **GroupSelection** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the ticket number: ");

int ticketNumber = sc.**nextInt**();

if (ticketNumber % 10 == 0){

**System**.out.**println**("Group Leader");

} else {

**System**.out.**println**("Group Member");

}

}

}

**Output:**

Enter the ticket number: 102930

Group Leader

**4. Ask user to enter age, and sex (M or F), then using following rules print their place of service.**

**a) if employee is female, then she will work only in urban areas.**

**b) if employee is a male and age is in between 20 to 40 then he may work in anywhere**

**c) if employee is male and age is in between 40 to 60 then he will work in urban areas only.**

**d) And any other input of age should print "ERROR".**

**Sample Input: 26, F**

**Sample Output: Urban Area**

**Sample Input: 30, M**

**Sample Output: Anywhere**

**Analysis:**

In this problem, we have to determine the place of service of an employee based on their age

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **PlaceOfService** {

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

**Scanner** scanner = new **Scanner**(**System**.in);

**System**.out.**print**("Enter age: ");

int age = scanner.**nextInt**();

**System**.out.**print**("Enter sex (M or F): ");

char sex = scanner.**next**().**charAt**(0);

sex = **Character**.**toUpperCase**(sex);

if (sex == 'F') {

**System**.out.**println**("Urban areas");

} else if (sex == 'M') {

if (age >= 20 && age <= 40) {

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

} else if (age > 40 && age <= 60) {

**System**.out.**println**("Urban areas");

} else {

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

}

} else {

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

}

}

}

**Output :**

Enter age: 45

Enter sex (M or F): M

Urban areas

**5. Write a program to display the grade of a student based on his/her marks**

**Marks Grade Marks Grade**

**>90 O 81 to 90 A**

**71 to 80 B 61 to 70 C**

**50 to 60 D < 50 F**

**Analysis:**

In this problem, we have to determine the grade of a student based on their marks. We will take the marks as input from the user and calculate the grade based on the following conditions:

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **StudentGrade** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter your marks: ");

int marks = sc.**nextInt**();

**String** grade = "";

if (marks > 90){

grade = "O";

} else if (marks >= 81 && marks <= 90){

grade = "A";

} else if (marks >= 71 && marks <= 80){

grade = "B";

} else if (marks >= 61 && marks <= 70){

grade = "C";

} else if (marks >= 50 && marks <= 60){

grade = "D";

} else {

grade = "E";

}

**System**.out.**println**("Grade: "+ grade);

}

}

**Output:**

Enter your marks: 90

Grade: A

**6. The policy followed by a company to process customer orders is given by the following rules: Suppose stock=100. If a customer order is less than or equal to that in stock and his credit is OK, supply his requirements. If his credit is not OK do not supply. Send him intimation. If his credit is OK but the item in stock is less than is order, supply the available stock and inform as ‘Out of stock. Balance will be refunded’.**

**Sample Input: 150, 100, ‘N’**

**Sample Output: 100 supplied**

**Sample Input: 150, 100, ‘Y’**

**Sample Output: Cannot supply**

**Sample Input: 150, 200, ‘N’**

**Sample Output: 150 supplied. Out of stock. Balance will be refunded.**

**Analysis:**

In this problem, we have to determine whether a customer order can be processed based on the stock availability and the customer's credit status. We will take the customer order, stock availability, and credit status as input from the user and process the order based on the following conditions:

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **CustomerOrder** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the customer order: ");

int customerOrder = sc.**nextInt**();

**System**.out.**print**("Enter the stock: ");

int stock = sc.**nextInt**();

**System**.out.**print**("Enter the credit status (Y/N): ");

char creditStatus = sc.**next**().**charAt**(0);

creditStatus = **Character**.**toUpperCase**(creditStatus);

if (customerOrder <= stock && creditStatus == 'Y'){

**System**.out.**println**(customerOrder + " supplied");

} else if (creditStatus == 'N'){

**System**.out.**println**("Cannot supply");

} else {

**System**.out.**println**(stock + " supplied. Out of stock. Balance will be refunded");

}

}

}

**Output:**

Enter the customer order: 150

Enter the stock: 200

Enter the credit status (Y/N): Y

Cannot supply

**7. During lockdown, ATMs were allowed to dispatch currency bill in specific way. If user enters the amount, which is not multiple of 500 transactions will be rejected.**

**Sample Input: 3400 (assume balance is 10000)**

**Sample Output: Please enter amount multiple of 500**

**Sample Input: 2000 (assume balance is 10000)**

**Sample Output: Valid amount, transaction in process**

**Analysis:**

In this problem, we have to determine whether an ATM transaction can be processed based on the amount entered by the user. We will take the amount as input from the user and check whether it is a multiple of 500 to determine whether the transaction can be processed.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **ATMDuringLockdown** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the amount: ");

int amount = sc.**nextInt**();

int balance = 10000;

if (amount % 500 == 0){

**System**.out.**println**("Valid amount, transaction in process");

} else {

**System**.out.**println**("Please enter amount multiple of 500");

}

}

}

**Output:**

Enter the amount: 3400

Please enter amount multiple of 500

**8. Create a program to display the "text" corresponding to a certain "numerical value", using the following equivalence:**

**a) 9,10 = Excellent**

**b) 7,8 = Notable**

**c) 6 = Good**

**d) 5 = Approved**

**e) 0-4 = Fail**

**f) >=10= Invalid**

**Sample Input: 9**

**Sample Output: Excellent**

**Sample Input: 3**

**Sample Output: Fail**

**Analysis:**

In this problem, we have to determine the text corresponding to a certain numerical value based on the given equivalence. We will take the numerical value as input from the user and display the corresponding text based on the following equivalence:

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **NumericValue** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the numerical value: ");

int numericalValue = sc.**nextInt**();

**String** text = "";

if (numericalValue == 9 || numericalValue == 10){

text = "Excellent";

} else if (numericalValue == 7 || numericalValue == 8){

text = "Notable";

} else if (numericalValue == 6){

text = "Good";

} else if (numericalValue == 5){

text = "Approved";

} else if (numericalValue >= 0 && numericalValue <= 4){

text = "Fail";

} else {

text = "Invalid";

}

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

}

}

**Output:**

Enter the numerical value: 7

Notable

**9. Write a program to perform the following:**

**a) Display the question “What is the unit of currency in India?”**

**b) Accept the answer**

**c) If the answer is wrong (other than Rupee) display “Try again!”.**

**d) If it is correct “Rupee” display the message “Answer is correct”.**

**e) If the user gives the correct answer in first two attempts the program will terminate.**

**f) If the user fails to provide correct answer in three attempts the program itself gives the answer.**

**Sample Input/Output:**

**What is the unit of currency in India? Rupee**

**Answer is correct**

**Sample Input/Output:**

**What is the unit of currency in India? Euro**

**Try again!**

**Rupee**

**Answer is correct**

**Sample Input/Output:**

**What is the unit of currency in India? Yen**

**Try again!**

**Pound**

**Try again!**

**Euro**

**Sorry it is Rupee**

**Analysis:**

In this problem, we have to ask the user a question and accept the answer. If the answer is correct, we will display a message saying "Answer is correct". If the answer is wrong, we will display "Try again!" and give the user three attempts to provide the correct answer. If the user fails to provide the correct answer in three attempts, we will display the correct answer.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **CurrencyUnit** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**String** correctAnswer = "Rupee";

int attempt = 0;

boolean isAnswerCorrect = false;

while (attempt < 3 && !isAnswerCorrect){

**System**.out.**print**("What is the unit of currency in India? ");

**String** answer = sc.**nextLine**();

if (answer.**equalsIgnoreCase**(correctAnswer)){

**System**.out.**println**("Answer is correct");

isAnswerCorrect = true;

} else {

**System**.out.**println**("Try again!");

}

attempt++;

}

if (!isAnswerCorrect){

**System**.out.**println**("Sorry it is Rupee");

}

}

}

**Output:**

What is the unit of currency in India? Euro

Try again!

What is the unit of currency in India? Pound

Try again!

What is the unit of currency in India? Rupee

Answer is correct

**10. Read N. Find the Sum of first N odd numbers.**

**Sample Input: 3**

**Sample Output: 9**

**Sample Input: 7**

**Sample Output: 49**

**Analysis:**

In this problem, we have to find the sum of N odd number

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **SumOfOddNumbers** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the number: ");

int **N** = sc.**nextInt**();

int sum = 0;

int count = 1;

for (int i = 1; i <= **N**; i++){

sum += count;

count += 2;

}

**System**.out.**println**("Sum of first "+ **N** +" odd numbers: "+ sum);

}

}

**Output:**

Enter the number: 5

Sum of first 5 odd numbers: 25

**11. Write a program to print all possible outcomes with two dices (6**

**sides with digits 1 to 6)**

**Expected Output:**

**(1,1)(1,2)(1,3)(1,4)(1,5)(1,6)**

**(2,1)(2,2)(2,3)(2,4)(2,5)(2,6)**

**(3,1)(3,2)(3,3)(3,4)(3,5)(3,6)**

**(4,1)(4,2)(4,3)(4,4)(4,5)(4,6)**

**(5,1)(5,2)(5,3)(5,4)(5,5)(5,6)**

**(6,1)(6,2)(6,3)(6,4)(6,5)(6,6)**

**Analysis:**

In this problem, we have to print all possible outcomes when two dice are rolled. Each die has 6 sides with digits from 1 to 6. We will print all possible combinations of the outcomes of the two dice rolls.

**Code:**

package com.self\_practice;

public class **TwoDicePossible** {

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

for(int i=1;i<=6; i++){

for(int j=1;j<=6;j++){

**System**.out.**print**("("+i+","+j+") ");

}

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

}

}

}

**Output:**

(1,1) (1,2) (1,3) (1,4) (1,5) (1,6)

(2,1) (2,2) (2,3) (2,4) (2,5) (2,6)

(3,1) (3,2) (3,3) (3,4) (3,5) (3,6)

(4,1) (4,2) (4,3) (4,4) (4,5) (4,6)

(5,1) (5,2) (5,3) (5,4) (5,5) (5,6)

(6,1) (6,2) (6,3) (6,4) (6,5) (6,6)

**12. Write a program to print the multiplication table from 1 to N (where N is a number entered by user).**

**Sample Input: 10**

**Sample Output:**

**1 2 3 4 5 6 7 8 9 10**

**2 4 6 8 10 12 14 16 18 20**

**3 6 9 12 15 18 21 24 27 30**

**4 8 12 16 20 24 28 32 36 40**

**5 10 15 20 25 30 35 40 45 50**

**6 12 18 24 30 36 42 48 54 60**

**7 14 21 28 35 42 49 56 63 70**

**8 16 24 32 40 48 56 64 72 80**

**9 18 27 36 45 54 63 72 81 90**

**10 20 30 40 50 60 70 80 90 100**

**Analysis:**

In this problem, we have to print the multiplication table from 1 to N, where N is a number entered by the user. We will take the number N as input from the user and print the multiplication table from 1 to N using nested loops.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **Tables** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter any number: ");

int table = sc.**nextInt**();

for(int i=1;i<=table;i++){

for(int j=1;j<=10;j++){

**System**.out.**format**("%3d ", i\*j);

}

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

}

}

}

**Output:**

1 2 3 4 5 6 7 8 9 10

2 4 6 8 10 12 14 16 18 20

3 6 9 12 15 18 21 24 27 30

4 8 12 16 20 24 28 32 36 40

5 10 15 20 25 30 35 40 45 50

6 12 18 24 30 36 42 48 54 60

7 14 21 28 35 42 49 56 63 70

8 16 24 32 40 48 56 64 72 80

9 18 27 36 45 54 63 72 81 90

10 20 30 40 50 60 70 80 90 100

**13. Write a program to print the following star pattern**

**Sample Input: 5**

**Sample Output:**

**\***

**\*\***

**\*\*\***

**\*\*\*\***

**\*\*\*\*\***

**Analysis:**

In this problem, we have to print a star pattern with increasing number of stars in each row. We will take the number of rows as input from the user and print the star pattern using nested loops.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **StarPatternRightAngleTriangle** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter any number: ");

int range = sc.**nextInt**();

for(int i=0;i<range;i++){

for(int j=0;j<=i;j++){

**System**.out.**print**("\* ");

}

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

}

}

}

**Output:**

**\***

**\* \***

**\* \* \***

**\* \* \* \***

**\* \* \* \* \***

**14. Write a program to print the following pattern**

**Sample Input: 5**

**Sample Output:**

**1**

**12**

**123**

**1234**

**12345**

**Analysis:**

In this problem, we have to print a pattern with increasing numbers in each row. We will take the number of rows as input from the user and print the pattern using nested loops.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **NumberPatternRightAngleTriangle** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter any number: ");

int range = sc.**nextInt**();

for(int i=0;i<range;i++){

for(int j=1;j<=i+1;j++){

**System**.out.**print**(j+" ");

}

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

}

}

}

**Output:**

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

**15. Write a program to print the following pattern**

**Sample Input: 5**

**Sample Output:**

**\*\*\*\*\***

**\*\*\*\***

**\*\*\***

**\*\***

**\***

**Analysis:**

In this problem, we have to print a star pattern with decreasing number of stars in each row. We will take the number of rows as input from the user and print the star pattern using nested loops.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **StarPatternInvertedRightAngleTriangle** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter any number: ");

int range = sc.**nextInt**();

for(int i=range;i>0;i--){

for(int j=0;j<i;j++){

**System**.out.**print**("\* ");

}

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

}

}

}

**Output:**

**\* \* \* \* \***

**\* \* \* \***

**\* \* \***

**\* \***

**\***

**16. A person pays tax 10% if his income is more than 2.5 lacs for the amount excess to 2.5 lacs. Person having annual income lesser than 2.5 lac is exempted from tax. Write a program to implement it.**

**Sample Input: 200000**

**Sample Output: You are exempted from tax**

**Sample Input: 400000**

**Sample Output: Tax amount is 15000**

**Analysis:**

In this problem, we have to calculate the tax amount based on the annual income of a person. If the annual income is more than 2.5 lacs, then the person has to pay a tax of 10% on the amount exceeding 2.5 lacs. If the annual income is less than or equal to 2.5 lacs, then the person is exempted from tax. We will take the annual income as input from the user and calculate the tax amount based on the given conditions.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **IncomeTax** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter your annual income: ");

int income = sc.**nextInt**();

int taxAmount = 0;

if (income > 250000){

taxAmount = (int) ((income - 250000) \* 0.10);

**System**.out.**println**("Tax amount is "+ taxAmount);

} else {

**System**.out.**println**("You are exempted from tax");

}

}

}

**Output:**

Enter your annual income: 300000

Tax amount is 5000

**17. Read total shopping amount purchased in the shop, and then apply the discount as per the following discount criteria, then find and print the final amount that must be paid by the customer after subtracting the discount amount:**

**Shopping Amount Discount%**

**5000 and Above 25%**

**1000 – 4999 10%**

**Below 1000 5%**

**Analysis:**

In this problem, we have to calculate the discount amount based on the total shopping amount purchased in the shop. We will take the total shopping amount as input from the user and calculate the discount amount based on the following criteria:

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **ShoppingDiscount** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the total shopping amount: ");

int shoppingAmount = sc.**nextInt**();

int discountAmount = 0;

if (shoppingAmount >= 5000){

discountAmount = (int) (shoppingAmount \* 0.25);

} else if (shoppingAmount >= 1000 && shoppingAmount <= 4999){

discountAmount = (int) (shoppingAmount \* 0.10);

}

**System**.out.**println**(

"Final amount to be paid: "+ (shoppingAmount - discountAmount)

);

}

}

**Output:**

Enter the total shopping amount: 6000

Final amount to be paid: 4500

**18. Write a program to read date of birth as input & calculate the age.**

**Sample Input: 4 7 1989**

**Sample Output: 33 years 2 months 3 days (May differ. This result is calculated on 14-Sep-2022)**

**Analysis:**

In this problem, we have to calculate the age of a person based on their date of birth. We will take the date of birth as input from the user and calculate the age based on the current date.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

import java.time.**LocalDate**;

import java.time.**Period**;

public class **CalculateAge** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter your date of birth (dd mm yyyy): ");

int day = sc.**nextInt**();

int month = sc.**nextInt**();

int year = sc.**nextInt**();

**LocalDate** birthDate = **LocalDate**.**of**(year, month, day);

**LocalDate** currentDate = **LocalDate**.**now**();

**Period** period = **Period**.**between**(birthDate, currentDate);

**System**.out.**println**(

"Age: "+ period.**getYears**()

+ " years " + period.**getMonths**()

+ " months " + period.**getDays**() + " days"

);

}

}

**Output:**

Enter your date of birth (dd mm yyyy): 4 7 1989

Age: 33 years 2 months 10 days

**19. Write a Java program to calculate frequency of digits in a number**

**Sample Input: 34593334**

**Sample Output:**

**3 occurs 4 times**

**4 occurs 2 times**

**5 occurs 1 times**

**9 occurs 1 times**

**Analysis:**

In this problem, we have to calculate the frequency of digits in a number. We will take the number as input from the user and calculate the frequency of each digit in the number.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **FrequencyOfDigits** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter any number: ");

int number = sc.**nextInt**();

int[] frequency = new int[10];

while (number > 0){

int digit = number % 10;

frequency[digit]++;

number /= 10;

}

for (int i=0;i<10;i++){

if (frequency[i] > 0){

**System**.out.**println**(

i + " occurs "

+ frequency[i] + " times"

);

}

}

}

}

**Output:**

Enter any number: 34593334

3 occurs 4 times

4 occurs 2 times

5 occurs 1 times

9 occurs 1 times

**20. To check whether the given number is Armstrong number or not. Armstrong number in 3-digit numbers is a number whose sum of cubes of each digit is equal to the number itself. For example: 153 = 1\*1\*1 + 5\*5\*5 + 3\*3\*3 // 153 is an Armstrong number.**

**Sample Input: 153**

**Sample Output: true**

**Sample Input: 121**

**Sample Output: false**

**Analysis:**

In this problem, we have to check whether a given number is an Armstrong number or not. An Armstrong number is a number that is equal to the sum of the cubes of its digits. We will take the number as input from the user and check whether it is an Armstrong number based on the given conditions.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **ArmstrongNumber** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter any number: ");

int number = sc.**nextInt**();

int temp = number;

int sum = 0;

while (number > 0){

int digit = number % 10;

sum += digit \* digit \* digit;

number /= 10;

}

**System**.out.**println**(temp == sum);

}

}

**Output:**

Enter any number: 153

true

**21. During lockdown, ATMs were allowed to dispatch currencies in specific way. If user enters the amount, which is not multiple of 500 transactions will be rejected. If amount is 500 exactly, then 5 currencies of 100 will be dispatched by machine.**

**Sample Input: 500**

**Sample Output: please dispatch 5 notes of 100**

**Sample Input: 300**

**Sample Output: please enter the amount multiple of 500**

**Analysis:**

In this problem, we have to determine whether an ATM transaction can be processed based on the amount entered by the user. We will take the amount as input from the user and check whether it is a multiple of 500. If the amount is exactly 500, then the machine will dispatch 5 notes of 100. Otherwise, the transaction will be rejected.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **ATMDuringLockdown** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the amount: ");

int amount = sc.**nextInt**();

if (amount % 500 == 0){

if (amount == 500){

**System**.out.**println**("please dispatch 5 notes of 100");

} else {

**System**.out.**println**("please enter the amount multiple of 500");

}

}

}

}

**Output:**

Enter the amount: 500

please dispatch 5 notes of 100

**22. A person pays tax 10% if his income is between 2.5 to 5 lacs. On the amount greater than that 20% for 5 to 12 lacs, for amount above 12 lacs 30%. Person will enter the amount input and will get tax amount as output.**

**Sample Input: 200000**

**Sample Output: You are exempted from tax**

**Sample Input: 40000**

**Sample Output: Tax amount is 15000**

**Sample Input: 800000**

**Sample Output: 85000**

**Sample Input: 1300000**

**Sample Output: 195000**

**Analysis:**

In this problem, we have to calculate the tax amount based on the annual income of a person. If the annual income is between 2.5 to 5 lacs, then the person has to pay a tax of 10%. If the annual income is between 5 to 12 lacs, then the person has to pay a tax of 20% on the amount exceeding 5 lacs. If the annual income is above 12 lacs, then the person has to pay a tax of 30% on the amount exceeding 12 lacs. We will take the annual income as input from the user and calculate the tax amount based on the given conditions.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **IncomeTax** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter your annual income: ");

int income = sc.**nextInt**();

int taxAmount = 0;

if (income > 250000 && income <= 500000){

taxAmount = (int) ((income - 250000) \* 0.10);

} else if (income > 500000 && income <= 1200000){

taxAmount = (int) ((income - 500000) \* 0.20 + 25000);

} else if (income > 1200000){

taxAmount = (int) ((income - 1200000) \* 0.30 + 125000);

}

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

}

}

**Output:**

Enter your annual income: 1300000

195000

**23. In an E-commerce website, there is an offer coupon for books worth Rs. 500 as a special promotion. There is also a separate discount for books, and it is 10%. Apply the max discount between coupon offer and individual discount for books in the final invoice. The user purchases the below book. Calculate the final bill amount to be paid by the user.**

**Harry potter book - 1 no. - Rs. 750**

**Analysis:**

In this problem, we have to calculate the final bill amount to be paid by the user for purchasing a book. The book costs Rs. 750, and there is a special promotion coupon worth Rs. 500. There is also a discount of 10% on the book. We have to apply the maximum discount between the coupon offer and the individual discount for books in the final invoice. We will calculate the final bill amount based on the given conditions.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **BookDiscount** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the book price: ");

int bookPrice = sc.**nextInt**();

int discount = 0;

if (bookPrice > 500){

discount = (int) (bookPrice \* 0.10);

} else {

discount = 500;

}

**System**.out.**println**("Final bill amount: "+ (bookPrice - discount));

}

}

**Output:**

Enter the book price: 750

Final bill amount: 675

**24. During lockdown, ATMs were allowed to dispatch currencies in specific way. If user enters the amount, which is not multiple of 500 transactions will be rejected. If amount is 500 exactly, then 5 currencies of 100 will be dispatched by machine. If amount is between 500 to 2000 you will get 500 in the form of 100s currencies, rest 500s currencies. If amount is greater than 2000, then 2000 currencies will be dispatched and rest will be dispatched in the denomination of 500, last 500 will be in the denomination of 100.**

**Sample Input: 400**

**Sample Output: please enter the amount multiple of 500**

**Sample Input: 500**

**Sample Output: 5 notes of 100**

**Sample Input: 1500**

**Sample Output: 5 notes on 100 & 2 notes on 500**

**Sample Input: 5000**

**Sample Output: 5 notes on 100, 2 notes of 2000, 1 notes of 500**

**Analysis:**

In this problem, we have to determine how an ATM transaction can be processed based on the amount entered by the user. We will take the amount as input from the user and check whether it is a multiple of 500. If the amount is exactly 500, then the machine will dispatch 5 notes of 100. If the amount is between 500 to 2000, then the machine will dispatch 500 in the form of 100s currencies and the rest in 500s currencies. If the amount is greater than 2000, then the machine will dispatch 2000 currencies and the rest in the denomination of 500, with the last 500 in the denomination of 100.

**Code:**

package com.self\_practice;

import java.util.**Scanner**;

public class **ATMDuringLockdown1** {

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

**Scanner** sc = new **Scanner**(**System**.in);

**System**.out.**print**("Enter the amount: ");

int amount = sc.**nextInt**();

int[] notes = {2000, 500, 100};

int count = 0, i = 0;

while (i < 3){

if (amount >= notes[i]){

count = amount % notes[i];

**System**.out.**println**(count + " notes on "+ notes[i]);

amount = amount % notes[i];

}

i++;

}

}

}

**Output:**

Enter the amount: 5000

5 notes on 100

2 notes on 500

2 notes on 2000