

```

/*
Author: Raul Martos
Date: 07/02/2020
*/
package pkraulmartos;

import java.text.DecimalFormat;
import java.util.Scanner;

public class M3UF2P17 {

    static Scanner keyboard = new Scanner(System.in);
    private static final String ANSI_PURPLE = "\u001B[35m";
    private static final String ANSI_CYAN = "\u001B[36m";
    private static final String ANSI_YELLOW = "\u001B[33m";

    public static void main(String[] args) {
        // TODO code application logic here
        System.out.println("Author: Raul Martos");
        System.out.println();

        int euro;
        String euro1;
        float euro2;
        String name;
        String results;
        DecimalFormat df = new DecimalFormat("0.00");
        int option = -1;
        keyboard.useDelimiter("\n");
    }
}

```

- 1- (int-Float) Función: Pide una cantidad en euro (int)y devuelve en monedaX (float) y visualiza:

```

case 1:
    System.out.print("How many euro? ");
    euro = keyboard.nextInt();
    float result = FunctionIntFloat(euro);
    System.out.print(euro + "€ " + " = " + df.format(result) + "$");
    System.out.println();
    break;
private static float FunctionIntFloat(int euro) {
    float result = 0;
    result = euro / 0.91f; //cast
    return result;
}

```

```

Choose an option
1
How many euro? 150
150€ = 164,84$

```

- 2- (int-double) Función: Pide una cantidad en euro(int) y devuelve en monedaX (double)y visualiza

```
case 2:
    System.out.print("How many euro? ");
    euro = keyboard.nextInt();
    double result2 = FunctionIntDouble(euro);
    System.out.println(euro + "€ " + " = " + df.format(result2) + "$");
    System.out.println();
    break;
private static double FunctionIntDouble(int euro) {
    double result2 = 0;
    result2 = euro / (double) 0.91;

    return result2;
}
```

```
Choose an option
2
How many euro? 150
150€ = 164,84$
```

- 3- (int-String) Función: Pide una cantidad en euro(int) y devuelve una string con el siguiente formato:

```
case 3:
    System.out.print("How many euro? ");
    euro = keyboard.nextInt();
    FunctionIntString(euro);
    System.out.println();
    break
private static String FunctionIntString(int euro) {
    String result3 = Integer.toString(euro);
    System.out.println(euro + "€ " + " = " + result3 + "$");

    return result3;
}
```

```
Choose an option
3
How many euro? 150
150 € = 150 $
```

- 4- (String-int) Función: Pide una cantidad en euro (string) y devuelve en monedaX y visualiza:

```
case 4:
    System.out.print("How many euro? ");
    euro1 = keyboard.next();
    int result4 = FunctionStringInt(euro1);
    System.out.println(euro1 + "€ " + " = " + result4 + "$");
    System.out.println();
    break;
private static int FunctionStringInt(String euro1) {
    int result4;
    result4 = Integer.parseInt(euro1);

    return result4;
}
```

```
Choose an option
4
How many euro? 150
150€ = 150$
```

5- (float-int) Función: Pide una cantidad en euro y devuelve en monedaX y visualiza:
000.00€ = 000x [usa cast]

```
case 5:
    System.out.print("How many euro? ");
    euro2 = keyboard.nextInt();
    int result5 = FunctionFloatInt(euro2);
    System.out.println(df.format(euro2) + "€ " + " = " + result5 + "$");
    System.out.println();
    break;
private static int FunctionFloatInt(float euro2) {
    int result5 = 0;
    result5 = (int) (euro2 / 0.91); //cast
    return result5;
}
```

```
Choose an option
5
How many euro? 150
150,00€ = 164$
```

6- (float-String) Función: Pide una cantidad en euro y devuelve una Sting con el siguiente formato:

```
case 6:
    System.out.println("How many euro? ");
    euro2 = keyboard.nextFloat();
    FunctionFloatString(euro2);
    System.out.println();
    break;
private static String FunctionFloatString(float euro2) {
    String result6 = Float.toString(euro2);
    System.out.println(euro2 + "€ " + " = " + result6 + "$");

    return result6;
}
```

```
Choose an option
€
How many euro?
150
150.0 € = 150.0 $
```

7- (char) procedimiento: pide una letra y visualiza su equivalente en int y binario

```
case 7:
    System.out.print("Letter: ");
    char letter = (keyboard.next()).charAt(0);
    FunctionChar(letter);
    break;
private static void FunctionChar(char letter) {
    String letterB = Integer.toBinaryString(letter);
    System.out.println(letter + " = " + letterB);
}
```

```
Choose an option
7
Letter: A
A = 1000001
```

8- (char-binary)- Función: Pide un nombre y devuelve su equivalente en binario

```
case 8:
    System.out.println("Enter one word: ");
    name = keyboard.next();
    results = FunctionCharBinary(name);
    System.out.println(results);
    System.out.println();
    break;
private static String FunctionCharBinary(String name) {
    String result = "";
    char ch = ' ';
    for (int i = 0; i < name.length(); i++) {
        ch = name.charAt(i);
        result = Integer.toBinaryString(ch) + " ";
    }
    return result;
}
```

Choose an option
8
Enter one word:
Raul
1010010

9- (int-char)- Función: Pide un número de 0 a 255 y visualiza su equivalente en char

```
case 9:
    int numChar;
    do {
        System.out.print("Give a number between 0 and 255: ");
        numChar = keyboard.nextInt();
    } while (numChar > 255 || 0 > numChar);
    String numconv = FunctionIntChar(numChar);
    System.out.println(numconv);
    System.out.println();
    break;
private static String FunctionIntChar(int numberintchar) {
    String aChar = new Character((char) numberintchar).toString();
    return aChar;
}
```

Choose an option
8
Give a number between 0 and 255: 140
"

10- (char-int)- Procedimiento: pon qué es la tabla de ASCII y visualízala: cada carácter con su equivalente numérico y en hexadecimal

```
case 10:
    FunctionCharInt();
    break;
private static void FunctionCharInt() {
    int Ncol = 3;
    int iCol=256/Ncol;
    for (int i = 0; i < 255; i++) {
        System.out.println(i + "\t" + Integer.toString(i, 16)
            + " = " + (char) i);
        System.out.print((i + iCol * 1) + " - " + (char) (i + iCol * 1)+"\t");
        System.out.print("\t" + (i + iCol * 2) + " - " + (char) (i + iCol * 2)+"\t");
    }
    System.out.println();
}
}
```

Choose an option

10

0	0 =	
85 - U	170 - ª 1	1 =□
86 - V	171 - « 2	2 =□
87 - W	172 - ¬ 3	3 =□
88 - X	173 - ¯ 4	4 =□
89 - Y	174 - ® 5	5 =□
90 - Z	175 - ¯ 6	6 =□
91 - [176 - ° 7	7 =□
92 - \	177 - ± 8	8 =□
93 -]	178 - º 9	9 =□
94 - ^	179 - º 10	a =□
95 - _	180 - ´ 11	b =□
96 - `	181 - µ 12	c =⬆
97 - a	182 - ¶ 13	d =□
98 - b	183 - · 14	e =□
99 - c	184 - ¸ 15	f =□
100 - d	185 - º 16	10 =□
101 - e	186 - ° 17	11 =□
102 - f	187 - » 18	12 =□
103 - g	188 - º 19	13 =□
104 - h	189 - º 20	14 =□
105 - i	190 - º 21	15 =□
106 - j	191 - º 22	16 =□
107 - k	192 - º 23	17 =□
108 - l	193 - º 24	18 =□
109 - m	194 - º 25	19 =□
110 - n	195 - º 26	1a =□