2023/2024(1)
EF234302 Object Oriented Programming

Lecture #3d

Exercises

Misbakhul Munir IRFAN SUBAKTI

司馬伊凡

Мисбакхул Мунир Ирфан Субакти

Ex1. Circle area: Problem

- The area *A* of the circle is computed by $\pi \cdot r^2$.
- Write a Java-program that makes this computation, initialize the variable `r` to 5 and print the result.
- Which types do you use for the variables *A* and `r`? (Hint: Use `Math.PI`).

Ex1. Circle area: Solution

```
public class Ex1 {
      /** The variable A and r are declared as double and A is computed
        *   from r as Math.PI * r * r and printed suitably.
        *   The program is test with r = 5.
        */
      public static void main(String[] args) {
       double a;
       double r;
       r = 5;
       a = Math.PI * r * r;
       System.out.println("The area of a circle with radius " + r +
                         " is " + a + ".");
      }
}
```

Ex2. Conversion: Problem

- Write a Java-program that converts masses given in the imperial system into kilograms. Make use of the following conversions:

1 ton	2240 pounds
1 hundredweight	112 pounds
1 quarter	28 pounds
1 stone	14 pounds
1 ounce	1/16 pounds
1 drachm	1/256 pounds
1 grain	1/7000 pounds
1 pound	0.45359237 kilograms

- A person’s weight corresponds to 11 stones and 6 pounds. Concretely the weight is stored by the two variables stones = 11; and pounds = 6; (all other variables such as tons are 0). Use your program to determine how many kilograms this is.

Ex2. Conversion: Solution (1)

```
public class Ex2 {
     /**
       * First the relationships are stored in variables:
       * 1 ton                  ~ 2240 pounds
       * 1 hundredweight        ~ 112 pounds
       * 1 quarter              ~ 28 pounds
       * 1 stone                ~ 14 pounds
       * 1 ounce                ~ 1/16 pounds
       * 1 drachm               ~ 1/256 pounds
       * 1 grain                ~ 1/7000 pounds
       * 1 pound                ~ 0.45359237 kilograms
       * Then the imperial weight is transformed into pounds and
       * stored in a variable  weightInPounds. Then this weight is
       * transformed in kilograms and the transformation is run for the 11 stones
       * and 6 pounds from the worksheet. Finally the result is suitably printed.
       */
```

Ex2. Conversion: Solution (2)

```
public static void main(String[] args) {
   double poundsPerTon               = 2240;
   double poundsPerHundredweight     = 112;
   double poundsPerQuarter           = 28;
   double poundsPerStone             = 14;
   double poundsPerOunce             = 1/16.0;
   double poundsPerDrachm            = 1/256.0;
   double poundsPerGrain             = 1/7000.0;
   double kilogramsPerPound          = 0.45359237;

   double tons = 0;
   double hundredweights = 0;
   double quarters = 0;
   double stones = 0;
   double pounds = 0;
   double ounces = 0;
   double drachms = 0;
   double grains = 0;
```

Ex2. Conversion: Solution (3)

```
         /* We compute first the pounds by converting the stones to pounds and
            add the pounds. Then we multiply by kilograms_per_pound.
          */
         stones = 11;
         pounds = 6;
         double weightInPounds =
                  tons * poundsPerTon +
                  hundredweights * poundsPerHundredweight +
                  quarters * poundsPerQuarter +
                  stones * poundsPerStone +
                  pounds +
                  ounces * poundsPerOunce +
                  drachms * poundsPerDrachm +
                  grains * poundsPerGrain;
         double weightInKilograms =  weightInPounds * kilogramsPerPound;
         System.out.println("A person with " + stones + " stones and " + pounds +
                  " pounds has a weight corresponding to " + weightInKilograms + " kilograms.");
          }
}
```

Ex3. Investment: Problem

- A capital of GBP 100 (£100) is invested at a fixed interest rate of 2.3 per cent.
- The interest is added to the capital at the end of each year.
- Use the formula to print the balance after each of the first 5 years and after 500 years.

$$\text{total} = \text{capital} * (1 + 0.01 * \text{interest rate})^{\text{years}}$$

Ex3. Investment: Solution (1)

```
public class Ex3 {
     /** In this exercise, the variables are declared and
       *  initialized, then the formula on the worksheet is
       *  translated to Java and the total capital computed after 1,
       *  2, 3, 4, 5, and 500 years. The code is repetitive (which is
       *  bad), since we have not introduced methods at this point in
       *  time yet.
       */
     public static void main(String[] args) {
        double capital = 100; // the initial capital in pounds
        double interestRate  = 2.3; // the interest rate per year
        int years = 1;   // the time for which the capital is invested in years.
        double total;  // the resulting capital after interest added.
        total = capital * Math.pow(1 + 0.01 * interestRate, years);
```

Ex3. Investment: Solution (2)

```
          System.out.println("After " + years + " years a capital of GBP " + capital +
                    " grows with an interest rate of " + interestRate + "\u0025 to GBP " +  Math.round(total));
                    // u0025 = percentSign (%)
          years = 2;
          total = capital * Math.pow(1 + 0.01 * interestRate, years);
          System.out.println("After " + years + " years a capital of GBP " + capital +
                    " grows with an interest rate of " + interestRate + "\u0025 to GBP " +  Math.round(total));
          years = 3;
          total = capital * Math.pow(1 + 0.01 * interestRate, years);
          System.out.println("After " + years + " years a capital of GBP " + capital +
                    " grows with an interest rate of " + interestRate + "\u0025 to GBP " +  Math.round(total));
          years = 4;
          total = capital * Math.pow(1 + 0.01 * interestRate, years);
          System.out.println("After " + years + " years a capital of GBP " + capital +
                    " grows with an interest rate of “ + interestRate + "\u0025 to GBP " +  Math.round(total));
          years = 5;
          total = capital * Math.pow(1 + 0.01 * interestRate, years);
          System.out.println("After " + years + " years a capital of GBP " + capital +
                    " grows with an interest rate of " + interestRate + "\u0025 to GBP " +  Math.round(total));
          years = 500;
          total = capital * Math.pow(1 + 0.01 * interestRate, years);
          System.out.println("After " + years + " years a capital of GBP " + capital +
                    " grows with an interest rate of “ + interestRate + "\u0025 to GBP " +  Math.round(total));
     }
}
```