JAVA ASSIGNMENT 1: Speed Converter

1. toMilesPerHour

Write a method called **toMilesPerHour** that has **1 parameter** of **type double** with the name **kilometersPerHour**. This method needs to return the rounded value of the calculation of type **long**.

If the parameter **kilometersPerHour** is **less than 0**, the method **toMilesPerHour** needs to **return -1** to indicate an **invalid value.**

Otherwise, if it is positive, **calculate the value of miles per hour**, **round it and return it**. For conversion and rounding, check the notes in the text below.

Examples of input/output:

* toMilesPerHour(1.5); → should **return** value 1
* toMilesPerHour(10.25); → should **return** value 6
* toMilesPerHour(-5.6); → should **return** value -1
* toMilesPerHour(25.42); → should **return** value 16
* toMilesPerHour(75.114); → should **return** value 47

2. printConversion

Write another method called **printConversion** with **1 parameter of type double** with the name **kilometersPerHour**.

This method should not return anything (void) and it needs to calculate **milesPerHour** from the **kilometersPerHour** parameter.

Then it needs to print a message in the format **"XX km/h = YY mi/h".**

XX represents the original value **kilometersPerHour**.

YY represents the rounded **milesPerHour** from the **kilometersPerHour** parameter.

If the parameter **kilometersPerHour** is < 0 then print the text **"Invalid Value".**

Examples of input/output:

* printConversion(1.5); → should **print** the following text (into the console - System.out): 1.5 km/h = 1 mi/h

* printConversion(10.25); → should **print** the following text (into the console - System.out): 10.25 km/h = 6 mi/h

* printConversion(-5.6); → should **print** the following text (into the console - System.out): Invalid Value

* printConversion(25.42); → should **print** the following text (into the console - System.out): 25.42 km/h = 16 mi/h

* printConversion(75.114); → should **print** the following text (into the console - System.out): 75.114 km/h = 47 mi/h

Use method Math.round to round the number of calculated miles per hour(double). The method round returns long.

How to use the method round and how it works?

The **Math.round()** is a built-in math method which returns the closest long to the argument. The result is rounded to an integer by adding 1/2, taking the floor of the result after adding 1/2, and typecasting the result to type long. The method returns the value of the argument rounded to the nearest int value.

USAGE EXAMPLE:

```
double number = 1.5;
long rounded = Math.round(number);
System.out.println("rounded= " + rounded);
System.out.println("with 3.9= " + Math.round(3.9));
System.out.println("with 4.5= " + Math.round(4.5));
int sum = 45;
int count = 10;
// typecasting so result is double e.g. double / int -> double
double average = (double) sum / count;
long roundedAverage = Math.round(average);
System.out.println("average= " + average);
System.out.println("roundedAverage= " + roundedAverage);
```

OUTPUT:

```
rounded= 2
with 3.9= 4
with 4.5= 5
average= 4.5
roundedAverage= 5
```

TIP: In the method **printConversion**, call the method **toMilesPerHour** instead of duplicating the code.

NOTE: All methods should be defined as **public static** like we have been doing so far in the course.

NOTE: 1 mile per hour is 1.609 kilometers per hour

NOTE: Do not add a **main** method to the solution code.