**44-542 Object-Oriented Programming**

**Functions as Return Types**

1. Create a project with a package named **functionreturntypes**.
2. Add one java main class to the package you created previously. The java file should be named **FunctionsAsReturnTypes**.
3. Method **main** is shown here. Most statements are commented out. You can copy and paste this code into your program file. As you add the methods described below, uncomment statements in **main** to test your code.

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

**// System.out.println(divideBy(3).apply(47));**

**// System.out.println(divideBy(2).apply(18));**

**// System.out.println(divideBy(17).apply(3));**

**// System.out.println(divideBy(5).apply(17));**

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

**//**

**// System.out.println(divideBy(3.0).apply(47.0));**

**// System.out.println(divideBy(2.0).apply(18.0));**

**// System.out.println(divideBy(17.0).apply(3.0));**

**// System.out.println(divideBy(5.0).apply(17.0));**

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

**//**

**// System.out.println(contains().apply("bearcats", "bear"));**

**// System.out.println(contains().apply("bearcats", "arc"));**

**// System.out.println(contains().apply("bearcats", "arce"));**

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

**//**

**// System.out.println(contains("bear").apply("bearcats"));**

**// System.out.println(contains("arc").apply("bearcats"));**

**// System.out.println(contains("arce").apply("bearcats"));**

**}**

Add the overloaded methods **divideBy** and **contains** as described below

1. **public static IntFunction divideBy(int divisor)**: This method returns the integer quotient when the argument to the function is divided by **divisor**.
2. **public static Function<Double, Double> divideBy(double divisor)**: This method returns a double value which is the result of dividing the argument supplied to the function by **divisor**.
3. **public static BiFunction<String, String, Boolean> contains()**: This method returns a function with two arguments of type **String**. The function returns true if the first string contains the second string, and false otherwise.
4. **public static Function<String, Boolean> contains(String str)**: This method returns a function with one argument of type **String**. The function returns true if the argument to the function contains **str** (the method parameter), and false otherwise.