TUTORIAL 5

Question 3 in the exam paper focuses in on a discussion of types.

1. A type is defined as an Algebraic Data Type (ADT) as follows:

   Type MyProductType = (byte, boolean, boolean)

   How many possible values can a data item of this type take?

2. A type is defined as an ADT as follows:

   Type MySumType = boolean | short

   where *short* represents a 16-bit integer. How many possible values can a data item of this type take?

3. An ADT is defined in Scala as follows:

```
sealed trait Pet
case class Cat(name: String) extends Pet
case class Fish(name: String, colour: String) extends Pet
case class Squid(name: String, age: Int) extends Pet
```

   Complete the function below that uses pattern matching to produce the results that follow

```
def sayHi(p: Pet): String =
  p match {
     <complete this>
  }

sayHi(Cat("Bob"))
res0: String = Meow Bob!

sayHi(Squid("Steve", 10))
res1: String = Hi Steve.

sayHi(Fish("Dory", "blue"))
res2: String = Hello Blue fishy Dory.
```

4. The documentation for the get method of Map provides the following information:

   def get(key: A): **Option**[B]

   Optionally returns the value associated with a key.

The following code defines a Map and uses get to try to access values by key.

```
val numbers = Map(1->"uno", 2->"dos", 3->"tres" , 4->"cuatro")
 val one = numbers.get(2)
 val five = numbers.get(5)
```

What are the values of the variables *one* and *five*?

How can you safely access and print the values of these variables?