Maths packages summary

Primitive Type

- Primitive type acts only as a type name
- Does not provide extra logic for calculation
- Type names are in lower case

What we need to focus on exam

- Short
- Integer
- Long
- Float
- Double
- Boolean

Initialization

```
public class MyProgram {
        public static void main(String[] args) {

            int a = 10;
            Integer intValue1 = 10;
            Integer intValue2 = new Integer(10);
            Integer intValue3 = a;
            Integer intValue4 = null;

        }
}
```

Wrapper to primitive type

- This process is called box and unbox
- Wrapper class can be set to null

Object level access

```
public class MyProgram {
        public static void main(String[] args) {

            Integer intValue1 = 10;
            int toIntValue = intValue.intValue();
            short toShortValue = intValue.shortValue();
            double toDoubleValue = intValue.doubleValue();
            long toLongValue = intValue.longValue();
            String toStringValue = intValue.toString();
        }
}
```

Wrapper object level access

- All wrapper class provides function to covert to other primitive type value
- When your conversion is invalid, e.g. max integer to short, it will be over flow. So it still follows the primitive type casting rules

Class level access

```
public class MyProgram {
        public static void main(String[] args) {

            int maxValue = Integer.MAX_VALUE;
            int minValue = Integer.MIN_VALUE
            int convertValue = Integer.valueOf(“123”);

        }
}
```

Wrapper class level access

- Wrapper class provides lots of useful class level access utility function itself
- Instead of object level access, class level functions provides good error handling.
- When a invalid string is get converted, error will be thrown

Maths packages

- A complete utility class, only provides class level access
- Provides tons of useful Maths operations
- We called these class pure utility classes

Exam related

- Math.random()
- Math.max(int a, int b)
- Math.min(int a, int b)
- Math.abs(int a)
- Math.round(float a)
- Math.floor(float a)
- Math.pow(double a, double b)