Difference between overload and override

Overload

If a class having more than one methods with the same names and having different arguments / parameters

Return type doesn’t matter. Only arguments matters in method overloading.

Ex:

**package** com.model;

**public** **class** AddDemoOfMetOverloading {

**public** **int** add (**int** num1, **int** num2) {

**return** num1+num2;

}

**public** **float** add(**float** num1, **float** num2) {

**return** num1+num2;

}

}

Overriding

The parent class will be having the abstract methods. Those methods will be defined in the child classes.

In the context of overriding the return type does matters. I.e, The return type of the child is same as to be as the parent.

Does access modifiers matters?

Parents can have equal or lower level access modifiers.

Whereas the child should have equal or higher level access modifiers.

Protected access modifier.

Protected access modifier will act as a private to other classes. Public to the inherited classes.

Marker interface

Martialling and Demartialling

String

String Literal

String var = “Hello”; //Created in string pool

String Object

String var = new String (“Hello”); //Created in heap. Can dereference after our use.

Strings are immutable (Cannot change once we declared)

To modify or update the String data we have

StringBuffer() (Creates extra spaces)

StringBuilder()

Exception Handling