|  |  |  |  |
| --- | --- | --- | --- |
| No. | Property | Method overloading | Method overriding |
| 1 | Method names | Must be same | Must be same |
| 2 | Argument types | Must be different(at least order) | Must be same(including order) |
| 3 | Method signature | Must be different | Must be same |
| 4 | Return type | No restrictions | Must be same but this rule is applicable until 1.4 version only. From 1.5v onwards co-variant return types are allowed. |
| 5 | Private, static and final methods | Can be overloaded | Can not be overridden |
| 6 | Access modifiers | No restrictions | We can’t reduce scope of access modifier but we can increase. |
| 7 | Throws clause | No restrictions | If child class throws any checked exceptions compulsory parent class method should throw the same checked exception are its parent otherwise we will get compile time error but there are no restrictions for unchecked exceptions. |
| 8 | Method resolution | Always take care by compiler based on reference type. | Always takes care by JVM based on Runtime object |
| 9 | Also known as | Compile time polymorphism or static polymorphism or early binding | Runtime polymorphism or dynamic binding or late binding. |

In overloading we have to check only method names (must be same) and argument types (must be different) except this the remaining like return types, access modifiers..etc are not required to check.

But in overriding everything we have to check like method names ,argument types, return types, access modifiers etc.