How many ways to create object in java

1. Using new keyword
2. Using Reflection Class.forName
3. Using clone()
4. Using Object deserialization
5. Using ClassLoader
6. Using Unsafe class (Does not support in Java 11)

**Using Reflection Class.forName()**

**MyObject object = (MyObject) Class.forName(“com.ddlab.rnd.MyObject”).newInstance();**

**Using Clone()**

MyObject myobject = new MyObject();

MyObject anotherObject = myObject.**clone()**;

**Using Object Deserialization**

ObjectInputStream oin = **new** ObjectInputStream(inStream);  
Emp emp = (Emp) oin.readObject();

***Using Classloader***

Class<Emp> klass =(Class<Emp>)Test.**class**.getClassLoader().**loadClass**(**"com.ddlab.rnd.type4.Emp"**);  
Emp emp = (Emp) **klass.getDeclaredConstructor().newInstance()**;  
System.***out***.println(emp);

Class<E>.newInstance() 🡸 Deprecated in java 11.

**Usage of Unsafe class**

**public static void** main(String[] args) **throws** Exception {  
 Emp emp1 = **new** Emp();  
 emp1.setName(**"DD"**);  
 Field filed = Unsafe.**class**.getDeclaredField(**"theUnsafe"**);  
 Unsafe unsafe = (Unsafe) filed.get(**null**);  
 Emp emp = (Emp) unsafe.allocateInstance(Emp.**class**);  
 System.***out***.println(emp.getName());  
}

Java BeanInfo Introsepctor provides utilities to get the setter and getter methods.

BeanInfo info = Introspector.*getBeanInfo*(emp1.getClass());  
PropertyDescriptor[] propDescriptors = info.getPropertyDescriptors();  
**for**(PropertyDescriptor propDesc: propDescriptors) {  
 Method readMethod = propDesc.getReadMethod();  
 Method writeMethod = propDesc.getWriteMethod();  
}