

# Generics with Wildcards

- In generic code, the question mark (?), called the wildcard, represents an unknown type.
- ? is super than Object class
- The wildcard is never used as a type in class or interface declaration.

Don't do `class A<?> { }`

If there is `class A<T> { }`

then you can create object with unknown type `A<?> a = new A<className>`

# Unbounded wildcard parameterized type

```
ArrayList<?> list = new ArrayList<Long>();
```

Later you can assign any other object to wildcarded reference variable

```
list = new ArrayList<Integer>();
```

# Upper bounded wildcards

```
ArrayList<? extends Number> list = new ArrayList<Long>();
```

You can assign any other object with type which is **subclass** of Number

```
list = new ArrayList<Integer>();
```

# Lower bounded wildcards

```
ArrayList<? super Integer> list = new ArrayList<Number>();
```

You can assign any other object with type which is **superclass** of Integer.

```
list = new ArrayList<Object>();
```