Recitation 3
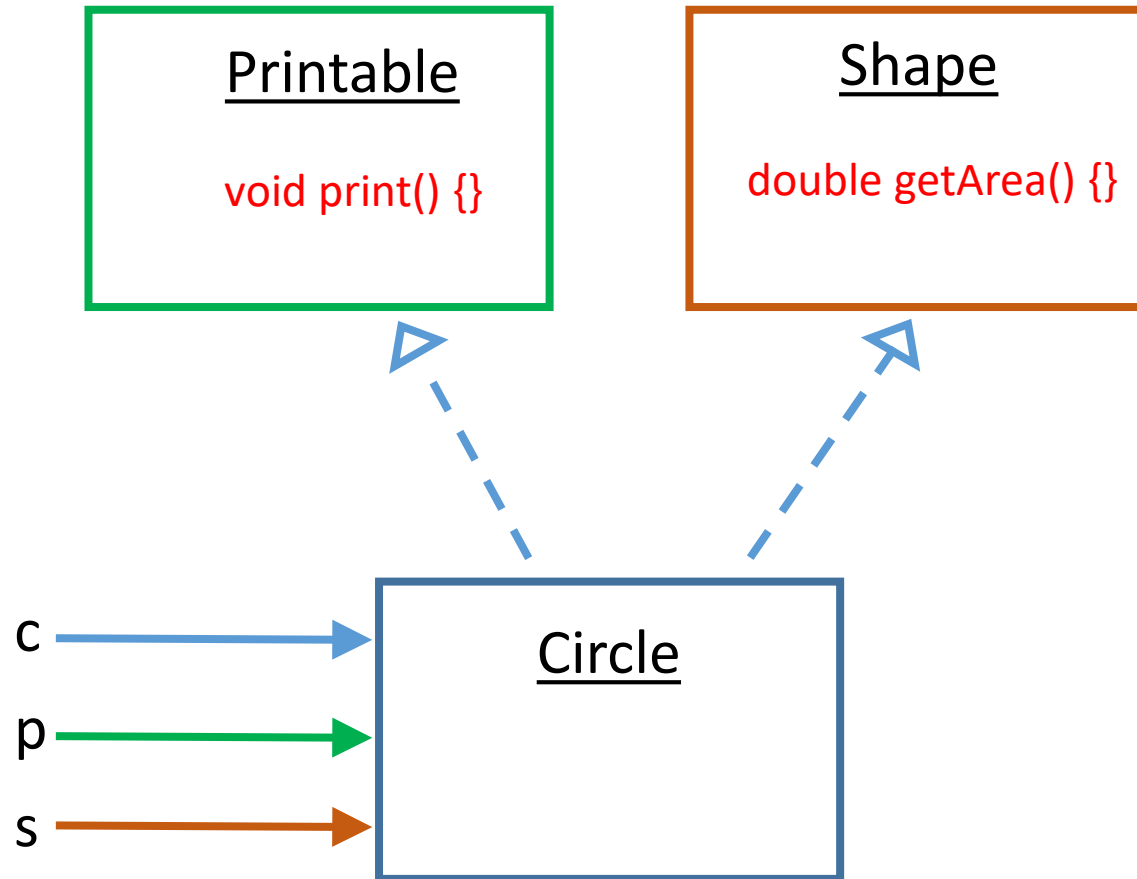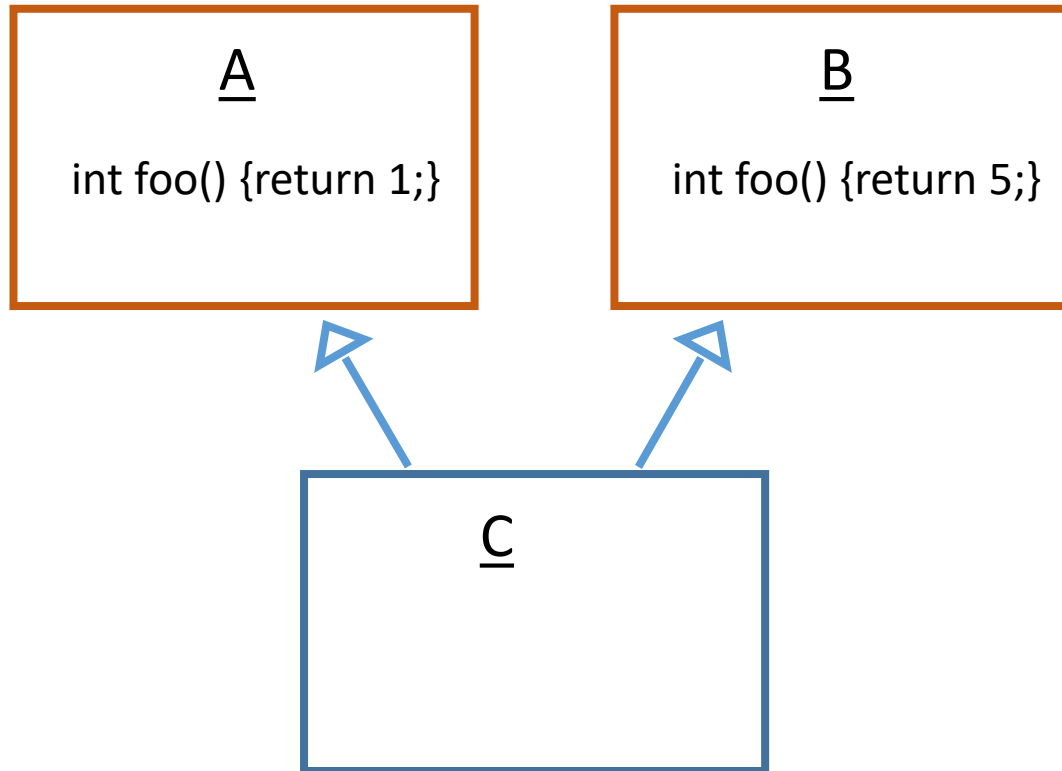
Q1(a) Interfaces are like super-classes
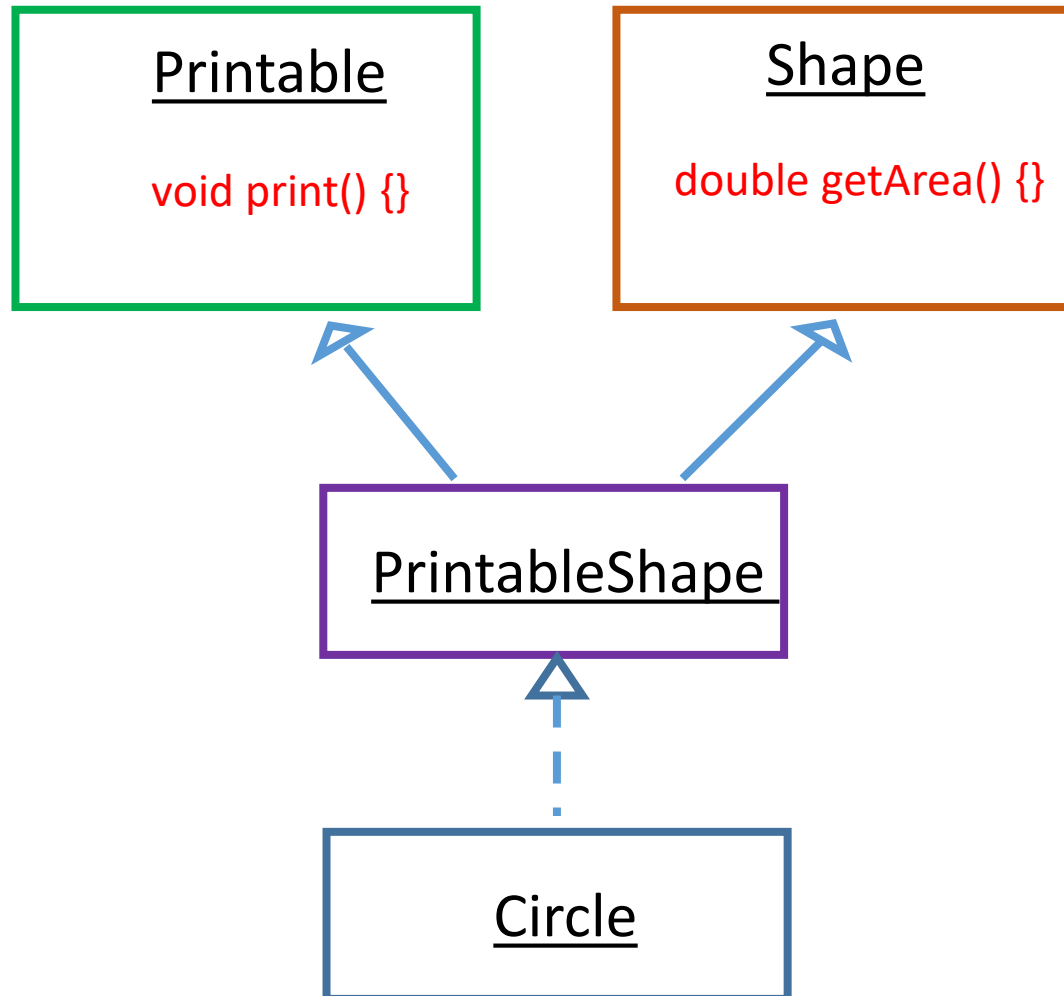
Q1(b) Java does not allow multiple inheritance (ie multiple parent classes)

A	B
int foo() {return 1;}	int foo() {return 5;}

C

new C().foo() *??*

Q1(c) Interfaces can be extended

Printable

void print() {}

Shape

double getArea() {}

PrintableShape

Circle

Q3

Liskov Substitution Principle (LSP):

The client of a class T expects the same behaviour if T was substituted with a subclass S of T

Q3

Liskov Substitution Principle (LSP):

> The client of a class *T* expects the same behaviour if *T* was substituted with a subclass *S* of *T*

Contract:

The client of FormattedText expects that toggling an instance will change its underline status: **true** to **false,** or **false** to **true**.

But this is not the case for PlainText.

Thus, PlainText violates LSP.