

## Basic Client/Server

```
interface Calculator {  
    long compute (in long data);  
};
```

---

```
class Calc extends CalculatorPOA {  
    public long compute (int data) {  
        // calculate x from data  
        return x;  
    }  
}
```

```
class Client {  
    private int value;  
    private Calculator calc1, calc2;  
  
    public void mainThread () {  
        value = calc1.compute(23) + calc2.compute(30);  
    }  
}
```

## Asynchronous Methods and Callbacks

```
interface ClientCB {  
    void result (in long answer);  
};
```

```
interface Calculator {  
    oneway void compute (in long data, in ClientCB cb);  
};
```

---

```
class Calc extends CalculatorPOA {  
    public void compute (int data, ClientCB cb) {  
        // calculate x  
        cb.result(x);  
    }  
}
```

```
class Client extends ClientCBPOA {  
    private int value=0;  
    private Calculator calc1, calc2;  
  
    public void mainThread () {  
        calc1.compute(23,this);  
        calc2.compute(30,this);  
    }  
  
    public void result (int answer) {  
        value = value + answer;  
    }  
}
```

## Race Conditions and Synchronization (A)

```
interface ClientCB {
    void result (in long answer);
};

interface Calculator {
    oneway void compute (in long data, in ClientCB cb);
};
```

---

```
class Calc extends CalculatorPOA {
    public void compute (int data, ClientCB cb) {
        // calculate x
        cb.result(x);
    }
}
```

```
class Client extends ClientCBPOA {
    private int value=0;
    private Calculator calc1, calc2;

    public synchronized void mainThread () {
        calc1.compute(23,this);
        calc2.compute(30,this);
    }

    public synchronized void result (int answer) {
        value = value + answer;
    }
}
```

*OR:*

```
public void result (int answer) {
    synchronized (value) {
        value = value + answer;
    }
}
```

## Race Conditions and Synchronization (B)

```
interface ClientCB {  
    void result (in long answer);  
};
```

```
interface Calculator {  
    oneway void compute (in long data, in ClientCB cb);  
};
```

---

```
class Calc extends CalculatorPOA {  
    public synchronized void compute (int data, ClientCB cb) {  
        // calculate x  
        cb.result(x);  
    }  
}
```

```
class Client extends ClientCBPOA {  
    private int value=0;  
    private Calculator calc1, calc2;  
  
    public synchronized void mainThread () {  
        calc1.compute(23,this);  
        calc2.compute(30,this);  
    }  
  
    public synchronized void result (int answer) {  
        value = value + answer;  
    }  
}
```

## Synchronization and Deadlock

```
interface ClientCB {  
    void result (in long answer);  
};
```

```
interface Calculator {  
    void compute (in long data, in ClientCB cb);  
};
```

---

```
class Calc extends CalculatorPOA {  
    public synchronized void compute (int data, ClientCB cb) {  
        // calculate x  
        cb.result(x);  
    }  
}
```

```
class Client extends ClientCBPOA {  
    private int value=0;  
    private Calculator calc1, calc2;  
  
    public synchronized void mainThread () {  
        calc1.compute(23,this);  
        calc2.compute(30,this);  
    }  
  
    public synchronized void result (int answer) {  
        value = value + answer;  
    }  
}
```

## Synchronization and Deadlock

```
interface ClientCB {  
    void result (in long answer);  
};
```

```
interface Calculator {  
    void compute (in long data, in ClientCB cb);  
};
```

---

```
class Calc extends CalculatorPOA {  
    public synchronized void compute (int data, ClientCB cb) {  
        // calculate x  
        cb.result(x);  
    }  
}
```

```
class Client extends ClientCBPOA {  
    private int value=0;  
    private Calculator calc1, calc2;  
  
    public void mainThread () {  
        calc1.compute(23,this);  
        calc2.compute(30,this);  
    }  
  
    public synchronized void result (int answer) {  
        value = value + answer;  
    }  
}
```