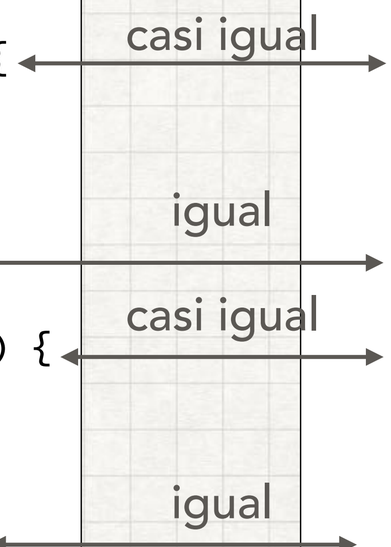
TEMPLATE METHOD PATTERN

```
public class SecretariaMaster {

  public boolean acepta (Solicitud solicitud) {

    if (!solicitudMasterCompleta(solicitud) ) {
      return false;
    }

    registra(solicitud);

    if (! tieneTituloUniversitario (solicitud)) {
      return false;
    }

    return hayPlaza(solicitud.getEstudios());

  }
...
}
```

```
public class SecretariaGrado {

  public boolean acepta (Solicitud solicitud) {

    if (! solicitudGradoCompleta(solicitud) ) {
      return false;
    }

    registra(solicitud);

    if (! tieneTituloPreUniversitario (solicitud)) {
      return false;
    }

    return hayPlaza(solicitud.getEstudios());

  }...
}
```

```java
public class SecretariaMaster {

  public boolean acepta (Solicitud solicitud) {

    if (!solicitudMasterCompleta(solicitud) ) {
      return false;
    }

    registra(solicitud);

    if (! tieneTituloUniversitario (solicitud)) {
      return false;
    }

    return hayPlaza(solicitud.getEstudios());

  }
...
}
```




```java
public class SecretariaGrado {

  public boolean acepta (Solicitud solicitud) {

    if (! solicitudGradoCompleta(solicitud) ) {
      return false;
    }

    registra(solicitud);

    if (! tieneTituloPreUniversitario (solicitud)) {
      return false;
    }

    return hayPlaza(solicitud.getEstudios());

  }
...
}
```

> Define the skeleton of an algorithm in an operation, deferring some steps to subclasses. Template Method lets subclasses redefine certain steps of an algorithm without changing the algorithm's structure.
>
> — *GoF*

```
public abstract class Secretaria {

  public boolean acepta (Solicitud solicitud) {

    if (! solicitudCompleta(solicitud) ) {
      return false;
    }

    registra(solicitud);

    if (! tieneTituloNecesario (solicitud)) {
      return false;
    }

    return hayPlaza(solicitud.getEstudios());

  }

  // implementacion comun de registra y hayPlaza

  abstract boolean solicitudCompleta (Solicitud solicitud);

  abstract boolean tieneTituloNecesario (Solicitud solicitud);
}
```

template method

```java
public class SecretariaMaster extends Secretaria {

  @Override
  boolean solicitudCompleta(Solicitud solicitud) {
    // implementación especifica para masters
  }

  @Override
  boolean tieneTituloNecesario(Solicitud solicitud) {
    // implementación especifica para masters
  }

}
```

TEMPLATE METHOD PATTERN