

# Programación Orientada a Objetos

```
var Carousel = {  
  init : function(options, el){  
    var base = this;
```

```
    base.$elem = $(el);
```

```
    // options passed via js override options passed via data attributes  
    base.options = $.extend({}, $.fn.owlCarousel.options, base.$elem.data(), options);
```

```
    base.userOptions = options;  
    base.loadContent();
```

```
  },
```

```
  loadContent : function(){  
    var base = this;
```

```
    if (typeof base.options.beforeInit === "function") {  
      base.options.beforeInit.apply(this, [base.$elem]);  
    }
```

```
    if (typeof base.options.jsonPath === "string") {  
      var url = base.options.jsonPath;
```

```
      function getData(data) {
```

```
        if (typeof base.options.jsonSuccess === "function") {  
          base.options.jsonSuccess.apply(this, [data]);
```



## Tener en cuenta...

- ¿Parejas conformadas?
- Laboratorio 1  
Viernes 30 de Enero
- Prácticas XP  
Bono Quiz \*
- Lecturas S3



## The Rules of Extreme Programming

### Planning

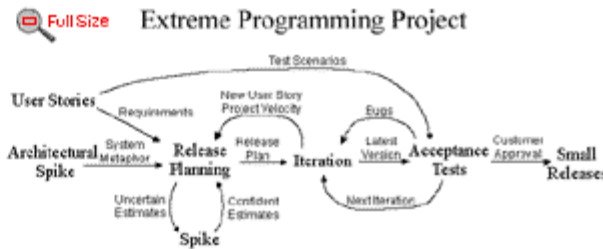
- User stories are written.
- Release planning creates the release schedule.
- Make frequent small releases.
- The project is divided into iterations.
- Iteration planning starts each iteration.

### Managing

- Give the team a dedicated open work space.
- Set a sustainable pace.
- A stand up meeting starts each day.
- The Project Velocity is measured.
- Move people around.
- Fix XP when it breaks.

### Designing

- Simplicity.
- Choose a system metaphor.
- Use CRC cards for design sessions.
- Create spike solutions to reduce risk.
- No functionality is added early.
- Refactor whenever and wherever possible.



### Coding

- The customer is always available.
- Code must be written to agreed standards.
- Code the unit test first.
- All production code is pair programmed.
- Only one pair integrates code at a time.
- Integrate often.
- Set up a dedicated integration computer.
- Use collective ownership.

### Testing

- All code must have unit tests.
- All code must pass all unit tests before it can be released.
- When a bug is found tests are created.
- Acceptance tests are run often and the score is published.

## Prácticas XP

- Contexto general sobre las prácticas XP (\*)
- Realizar material de apoyo (máximo 5 diapositivas incluyendo bibliografía)
- Moodle + otras fuentes
- Socializar en el laboratorio

# Prácticas XP

## Planning

- User stories are written.
- Release planning creates the release schedule.
- Make frequent small releases.
- The project is divided into iterations.
- Iteration planning starts each iteration.

- Exposición 5 minutos de:
  - ✓ Contexto general sobre las prácticas XP
  - ✓ ¿Qué propone?
  - ✓ ¿Para qué se utiliza?
  - ✓ ¿Cómo la usarían en POOB?

¿Voluntario 1?

# Prácticas XP

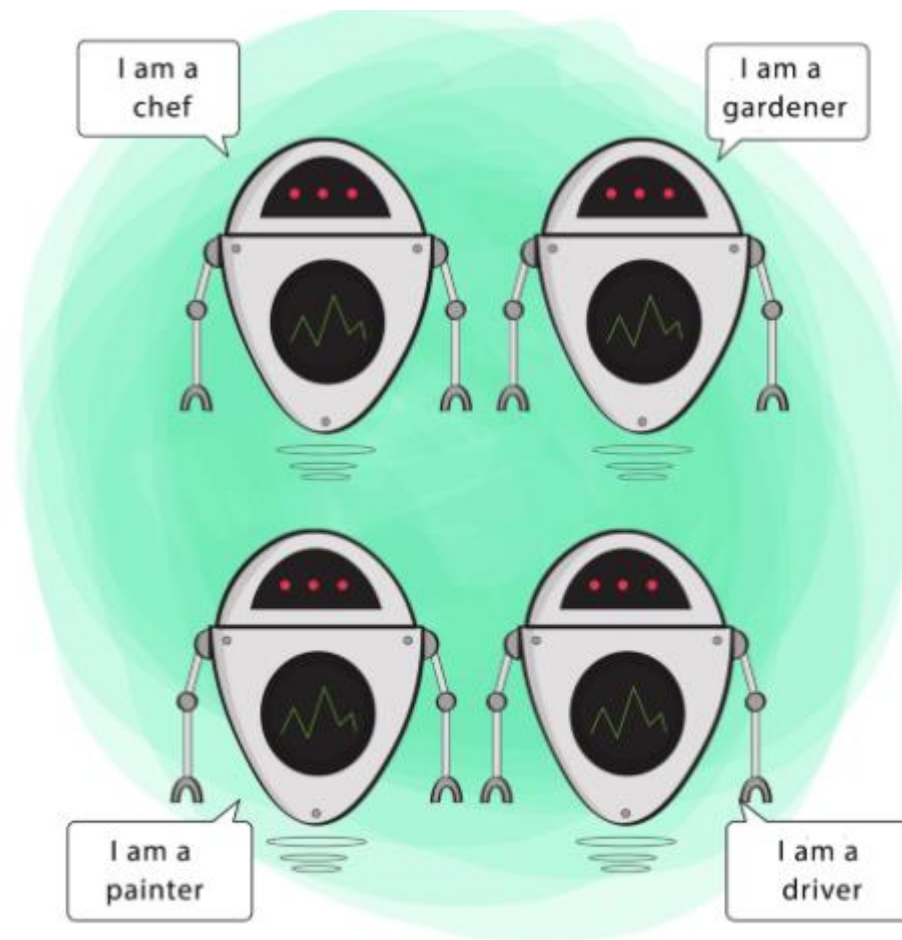
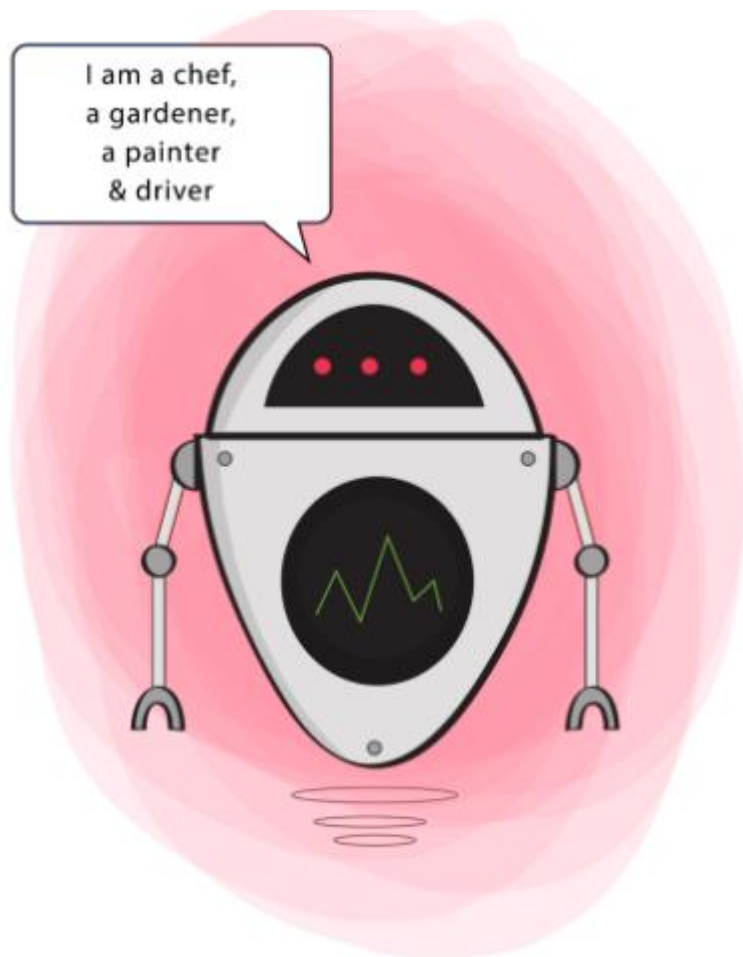
## Coding

- The customer is always available.
- Code must be written to agreed standards.
- Code the unit test first.
- All production code is pair programmed.
- Only one pair integrates code at a time.
- Integrate often.
- Set up a dedicated integration computer.
- Use collective ownership.

- Exposición 5 minutos de:
  - ✓ ¿Qué propone?
  - ✓ ¿Para qué se utiliza?
  - ✓ ¿Cómo la usarían en POOB?

¿Voluntario 2?

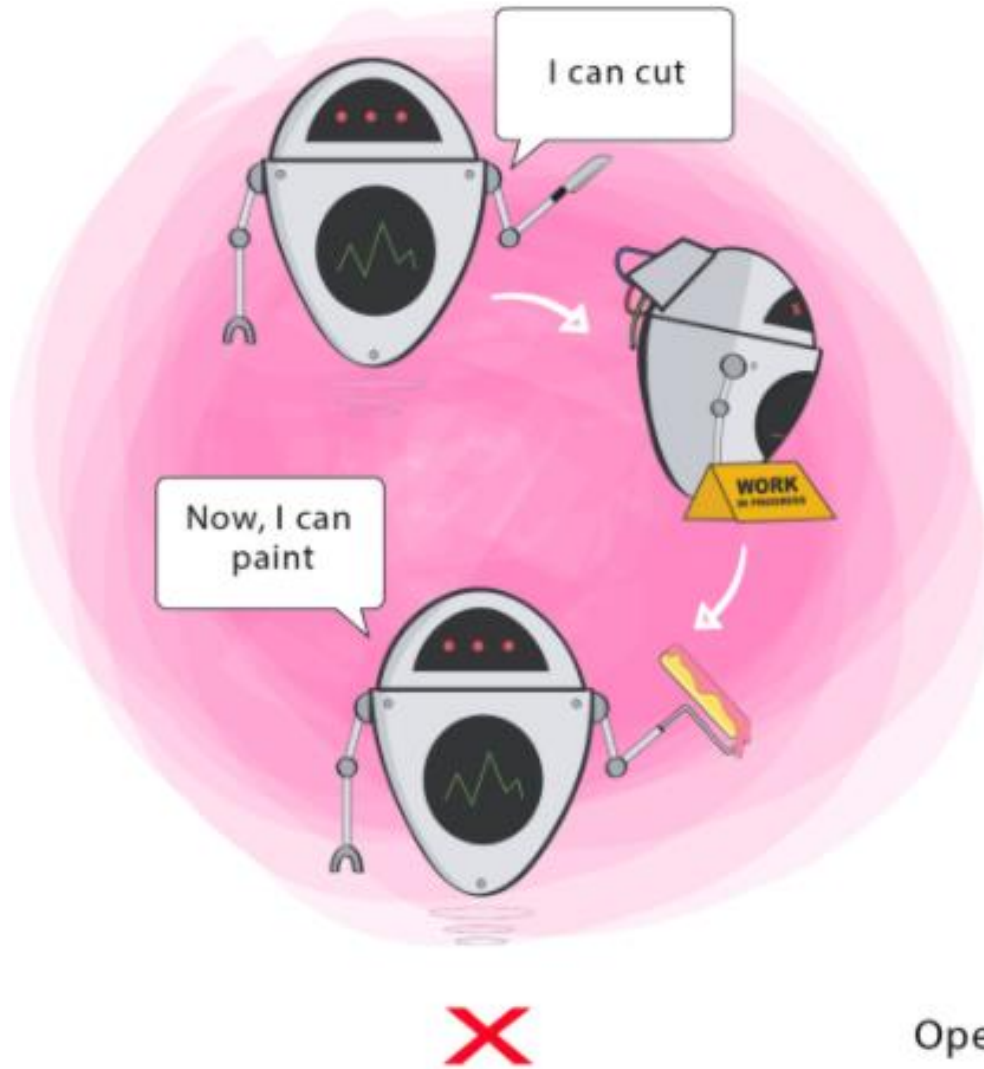
# S



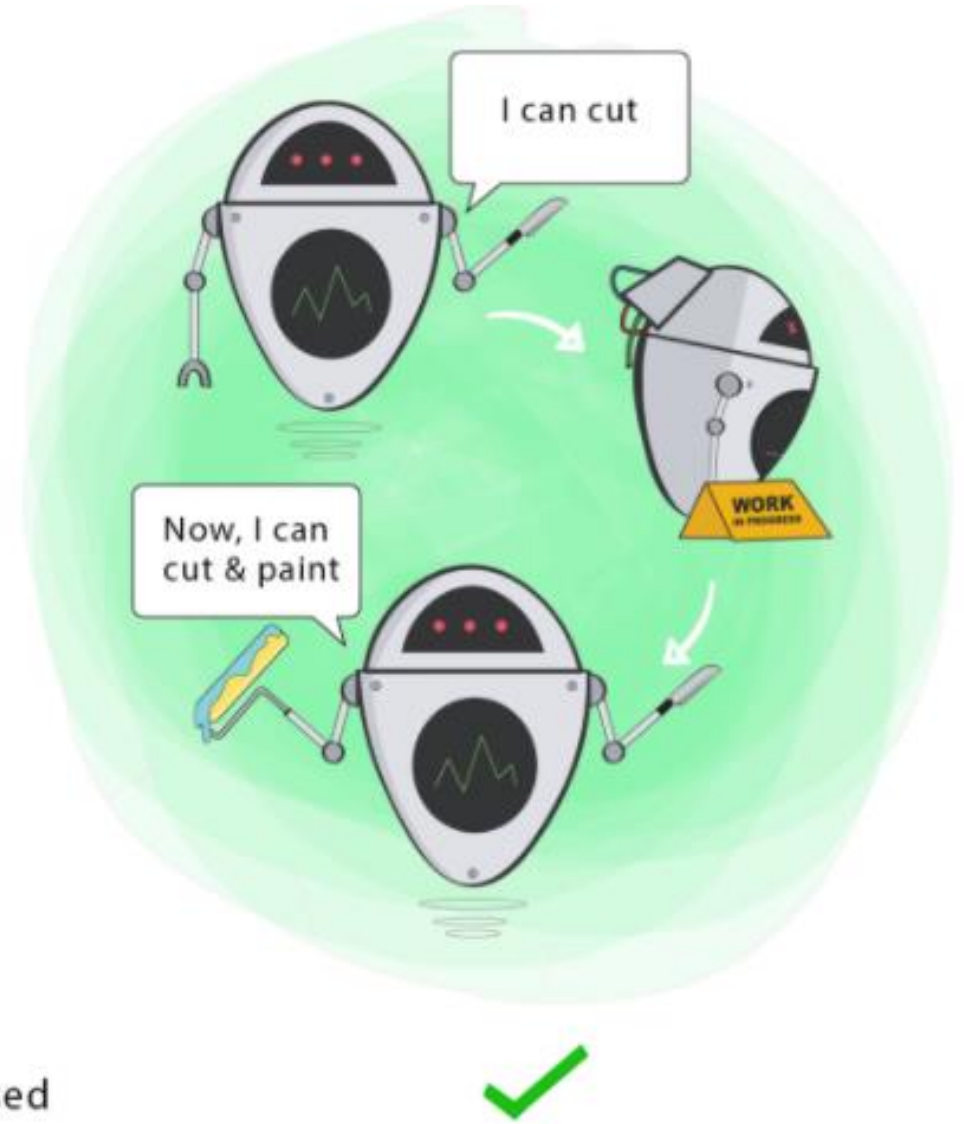
Single Responsibility



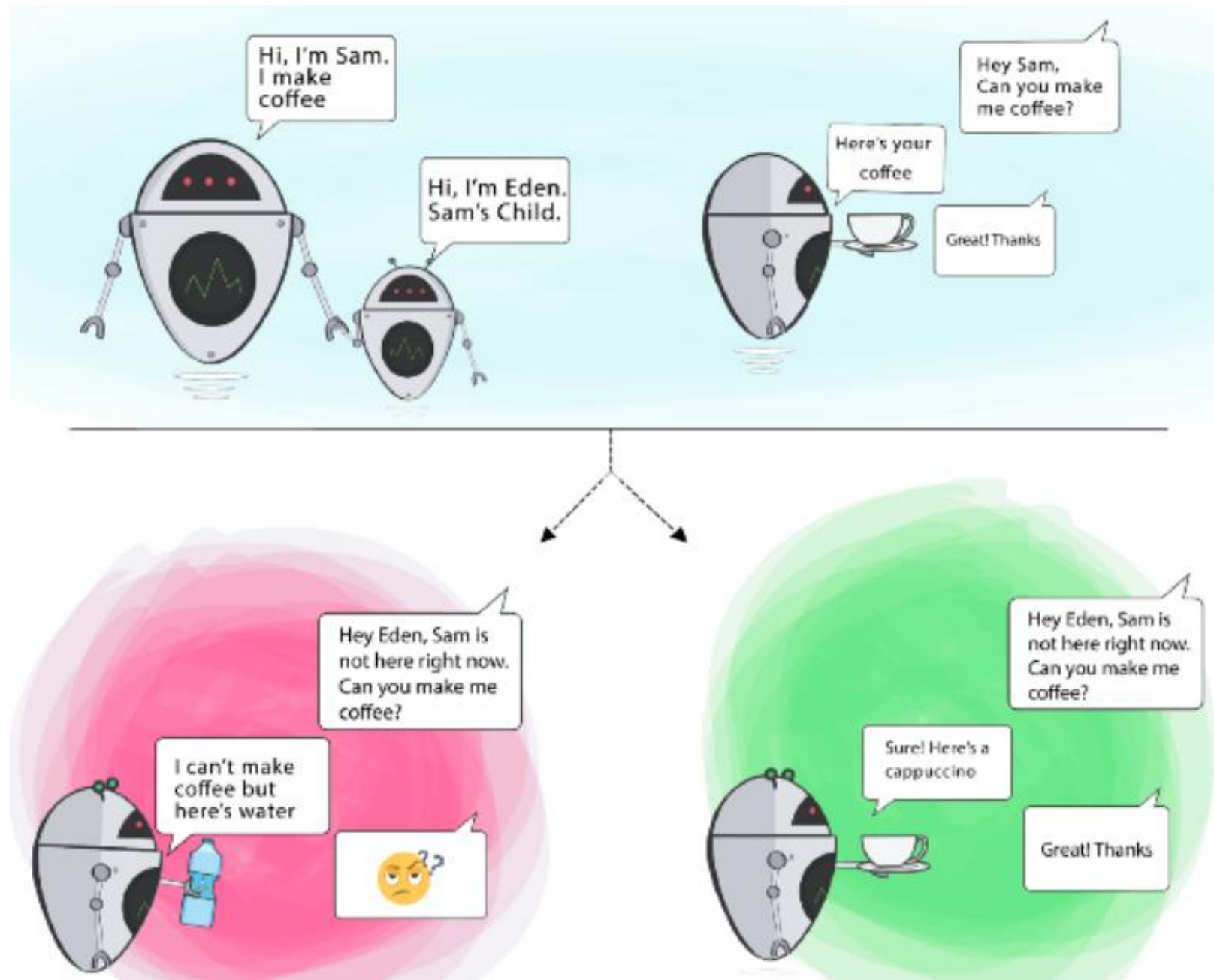
# O



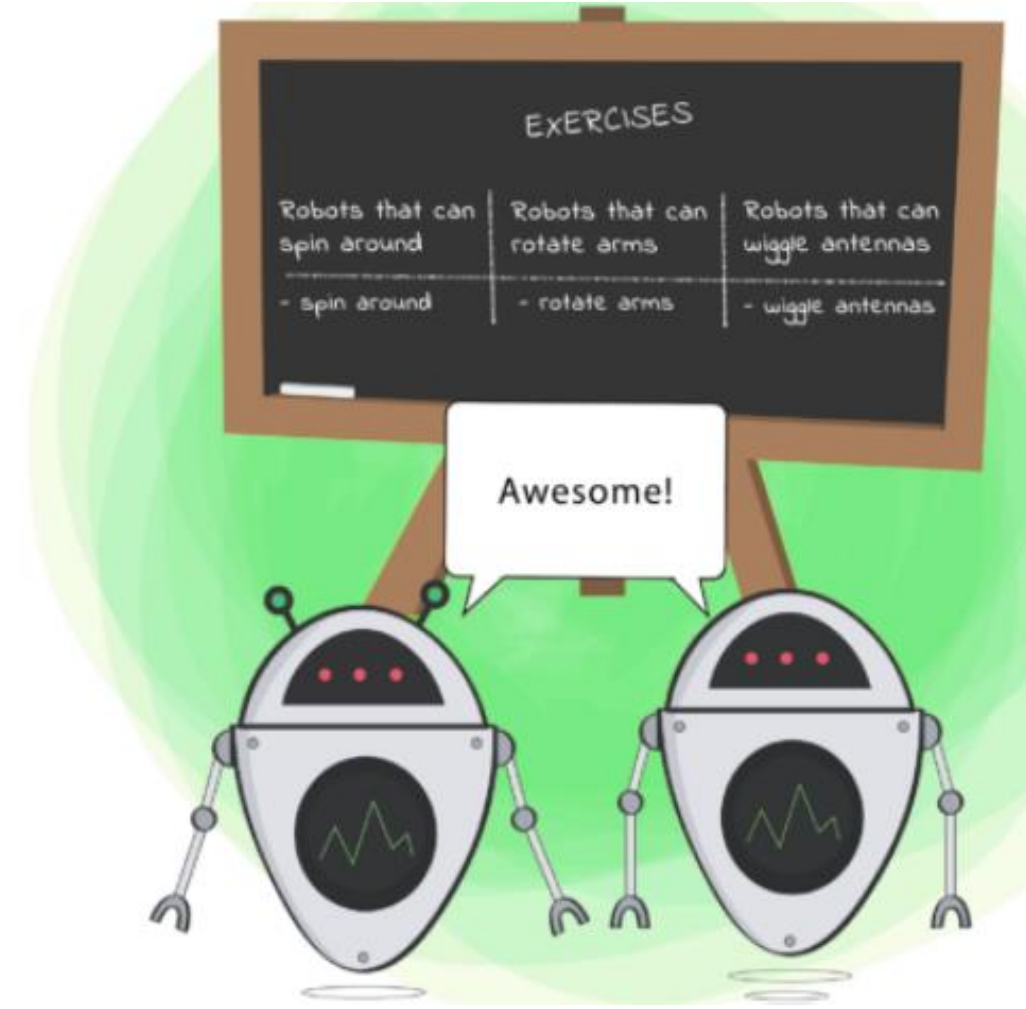
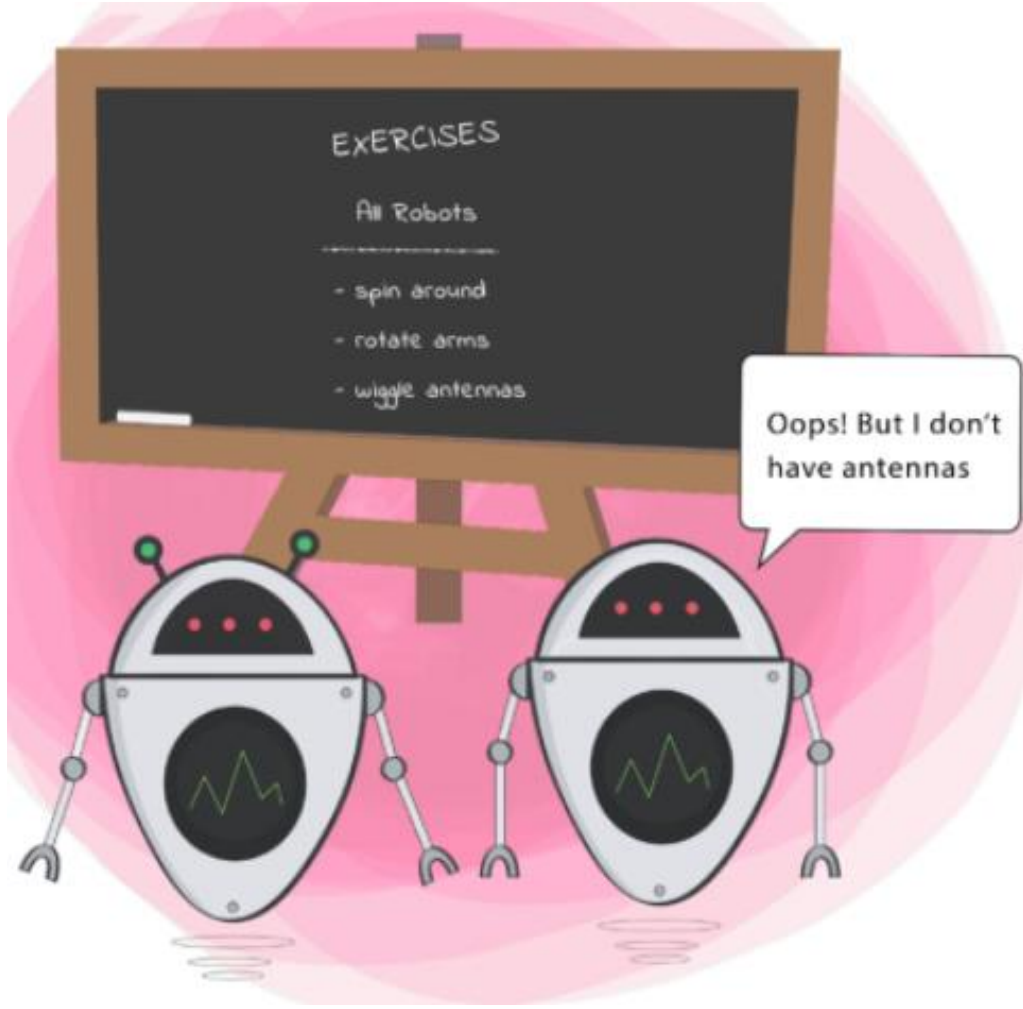
Open-Closed



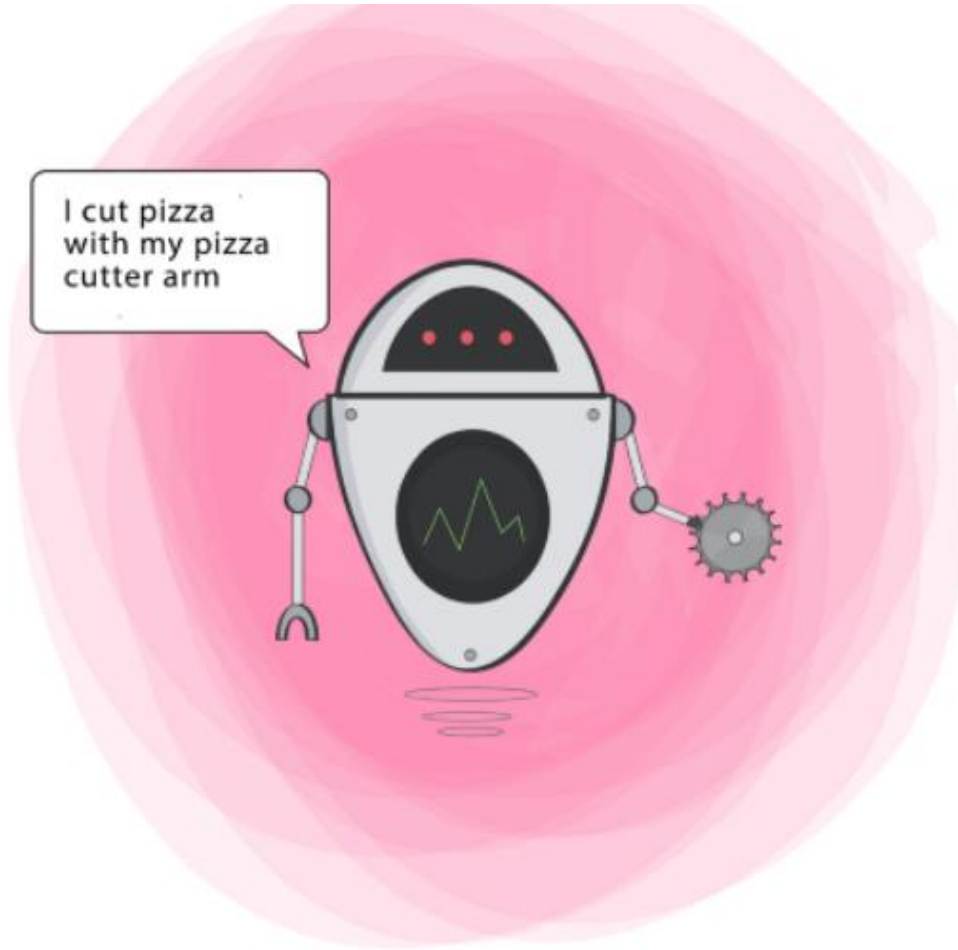
L



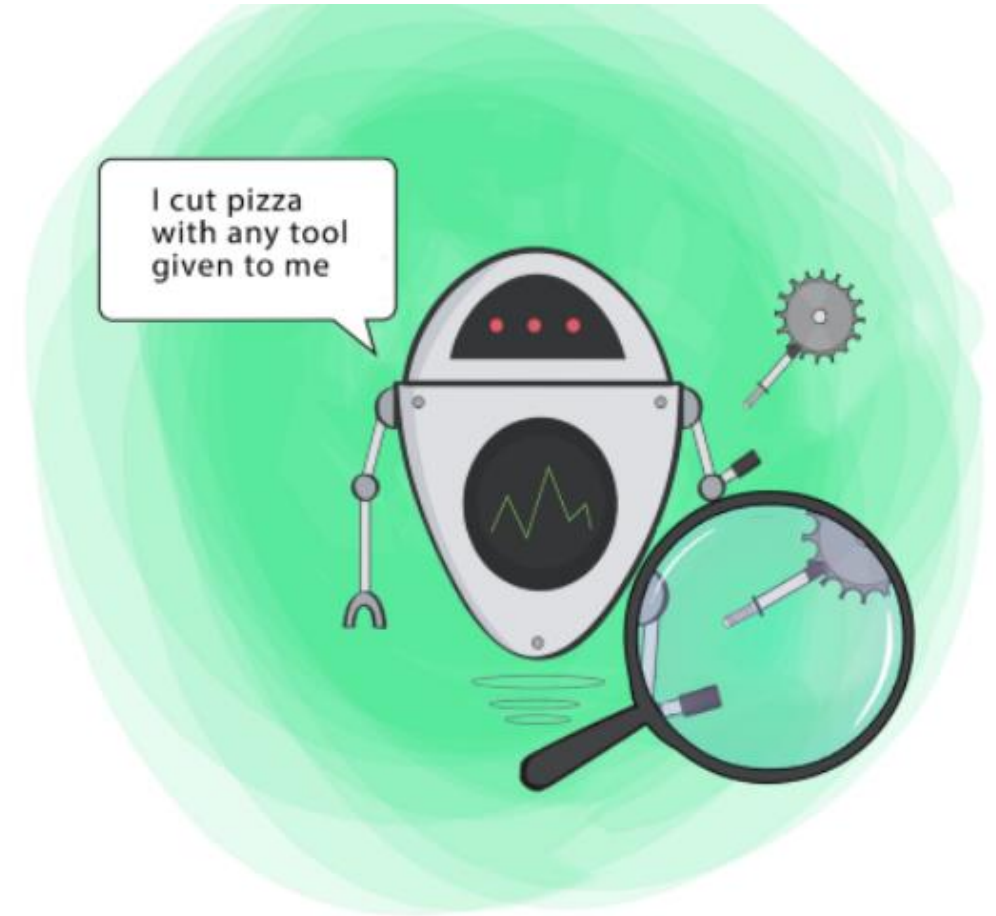




# D



✗



✓

Dependency Inversion