# Métodos iRobot_Framework

## Métodos principales

Lista de comandos recomendados para usar en la práctica 3:

- int connect();
- void start();
- void full();
- void drive(int speed, int radius);
- void driveDirect(int rightVelocity, int leftVelocity);
- int  updateSensor(char sensorId);
- void setVerbosity(int flag_debug);

## Códigos de sensores para el método *updateSensor*

```
namespace iRobotSensors{
    static const  char BUMPERS_AND_WHEELDROPS = (char) 7;
    static const  char WALL = (char) 8;
    static const  char CLIFFLEFT = (char) 9;
    static const  char CLIFFFRONTLEFT = (char) 10;
    static const  char CLIFFFRONTRIGHT = (char) 11;
    static const  char CLIFFRIGHT = (char) 12;
    static const  char VIRTUALWALL = (char) 13;
    static const  char MOTOR_OVERCURRENTS = (char) 14;
    static const  char IRBYTE = (char) 17;
    static const  char BUTTONS = (char) 18;
    static const  char DISTANCE = (char) 19;
    static const  char ANGLE = (char) 20;
    static const  char CHARGINGSTATE = (char) 21;
    static const  char VOLTAGE = (char) 22;
    static const  char CURRENT = (char) 23;
    static const  char TEMPERATURE = (char) 24;
    static const  char CHARGE = (char) 25;
    static const  char CAPACITY = (char) 26;
    static const  char WALLSIGNAL = (char) 27;
    static const  char CLIFFLEFTSIGNAL = (char) 28;
    static const  char CLIFFFRONTLEFTSIGNAL = (char) 29;
    static const  char CLIFFFRONTRIGHTSIGNAL = (char) 30;
    static const  char CLIFFRIGHTSIGNAL = (char) 31;
    static const  char USERDIGITAL = (char) 32;
    static const  char USERANALOG = (char) 33;
    static const  char CHARGINGSOURCES = (char) 34;
    static const  char OIMODE = (char) 35;
    static const  char SONGNUMBER = (char) 36;
    static const  char SONGPLAYING = (char) 37;
    static const  char STREAMPACKETS = (char) 38;
    static const  char VELOCITY = (char) 39;
    static const  char RADIUS = (char) 40;
    static const  char RIGHTVELOCITY = (char) 41;
    static const  char LEFTVELOCITY = (char) 42;
}
```

## Otros métodos disponibles

Otros comandos disponibles:
- void leds(int ledBit, int ledColor, int ledIntensity);
- void digitalOutputs(int outputBits);
- void lowSideDrivers(char outputBit);
- void pwmLowSideDrivers(int driver2,int driver1, int driver0);
- void song (int songNumber, int songSize,   char *song);
- void playSong(int songNumber);
- void baud(char code);
- void control();
- void safe();
- void stream(char* sensorIdList, int size);
- int  queryList(char * sensorIdList, int size);
- void PauseResumeStream(bool bol);
- void sendIr(int data);
- void demo(char code);
- void spot();
- void cover();
- void coverAndDock();

## Métodos para usar con precaución
Evitaremos el uso de las funciones que programan scripts ya que pueden bloquear el robot si no se usan adecuadamente.
- void script(int *commandList, int size);
- void playScript();
- void showScript();
- void waitTime(int seconds);
- void waitDistance(int mm);
- void waitAngle(int degrees);
- void waitEvent(int eventId);