

# First-class functions

```
function getAge() {  
    return 28;  
};
```

```
getAge();
```

or

```
const getAge = function() {  
    return 28;  
};
```

```
getAge();
```

or

```
const getAge = () => 28;  
getAge();
```

Anonymous functions will be very  
useful for object methods and  
callback methods

# Prototype-Based Object-Oriented

```
// defining a constructor
function Person(name) {
    this.name = name;
}

// adding a method
Person.prototype.getName = function() {
    return(this.name);
};

// creating an object
const p = new Person('Mariam');
console.log(p.getName());
console.log(p.constructor.name);
console.log(p instanceof Person);
```