



Apache
NetBeans

#25 E. REPETITIVAS

CICLO DO-WHILE



Java™

```
default.rb
# Scientist: Result
# An array of candidate observations.
attr_reader :candidates
# The control observation to which the rest are compared.
attr_reader :control
# An experiment.
attr_reader :experiment
# An array of observations which didn't match the control.
attr_reader :ignored
# An array of observations which didn't match the control.
attr_reader :mismatched
# An array of observations in execution order.
attr_reader :observations
# Internal: Create a new result.
#
# experiment - the Experiment this result is for
# observations - an Array of Observations, in execution order
# control - the control Observation
#
def initialize(experiment, observations = [], control = nil)
  @experiment = experiment
  @observations = observations
  @control = control
  @candidates = observations - [control]
  evaluate_candidates
end

freeze
end

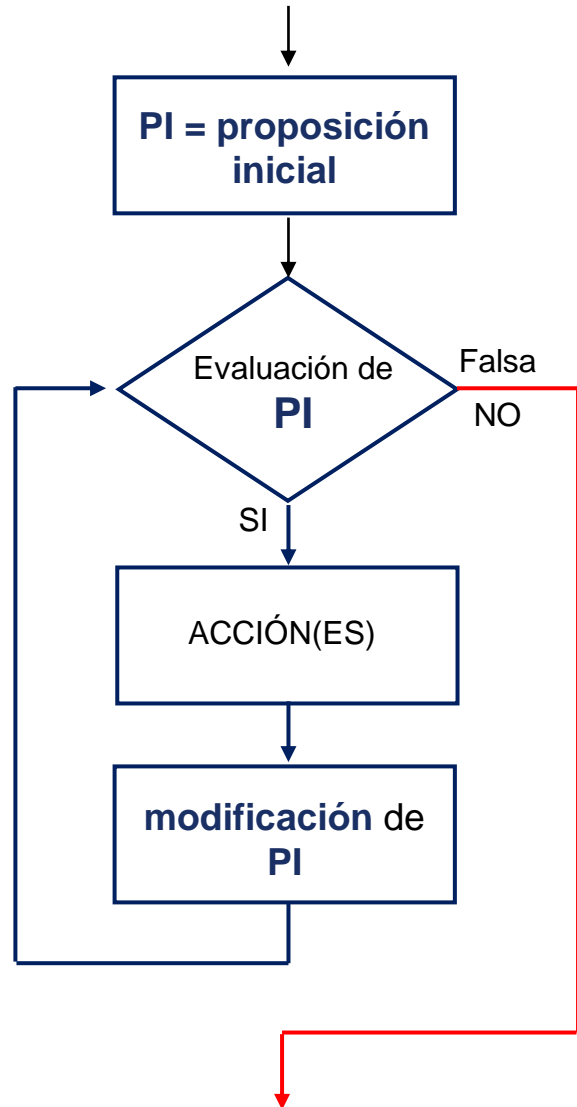
# Public: the experiment's context
def context
  experiment.context
end

# Public: the name of the experiment
def experiment_name
```

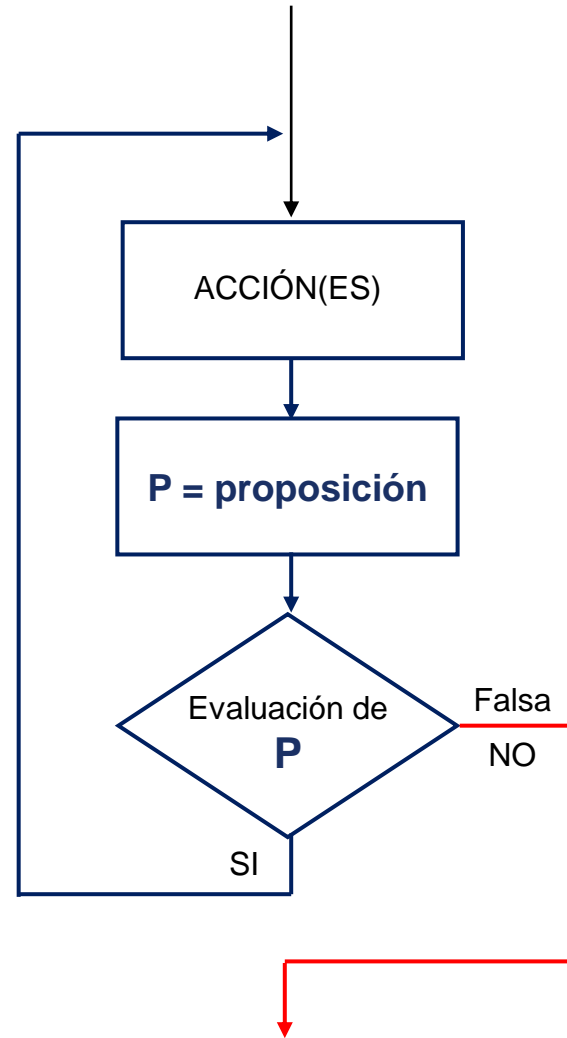
PROGRAMA FÁCIL CON JAVA



Ciclo while



Ciclo do while



do {

acción(es)

proposición P

} while (condición con P)

Ejemplo 3.10

Construye el programa en Java que obtenga lo que debe pagar un cliente por la compra de varios productos en una tienda. La captura termina cuando el usuario introduce un cero.

```
import java.util.Scanner;

public class Ejemplo310 {

    public static void main(String[] args) {
        Scanner Leer = new Scanner(System.in);
        float costo=0, total=0;
        do{
            total += costo;
            System.out.print("costo del producto: ");
            costo = Leer.nextFloat();    //Proposición
        }while(costo!=0);    //Evaluación de la Proposición
        System.out.println("Total = $" +total);
    }
}
```

Ejemplo 3.11

Construye el programa en Java que lea un indeterminado número de calificaciones, y que obtenga la cantidad de calificaciones leídas. Para terminar de leer se debe introducir una calificación con -1.

```
import java.util.Scanner;

public class Ejemplo311 {

    public static void main(String[] args) {
        Scanner Leer = new Scanner(System.in);
        int calif=0, i=-1;
        do{
            i++;
            System.out.print("Calificación: ");
            calif = Leer.nextInt();    //Proposición
        }while(calif!=-1);    //Evaluación de la Proposición
        System.out.println("Total de calificaciones "+i );
    }
}
```

Ejemplo 3.12

Construye el programa en Java que muestre N veces el texto “Fundamentos de Programación”.

```
import java.util.Scanner;
```

```
public class Ejemplo312 {
```

```
    public static void main(String[] args) {  
        Scanner Leer = new Scanner(System.in);  
        int i=1, N;  
        System.out.print("¿Cuántas veces se mostrará la materia?");  
        N = Leer.nextInt();  
        do{  
            System.out.println(i+".-Fundamentos de Programación");  
            i++;    //Proposición  
        }while(i<=N);    //Evaluación de la Proposición  
    }  
}
```