



UNIVERSIDAD CÉSAR VALLEJO

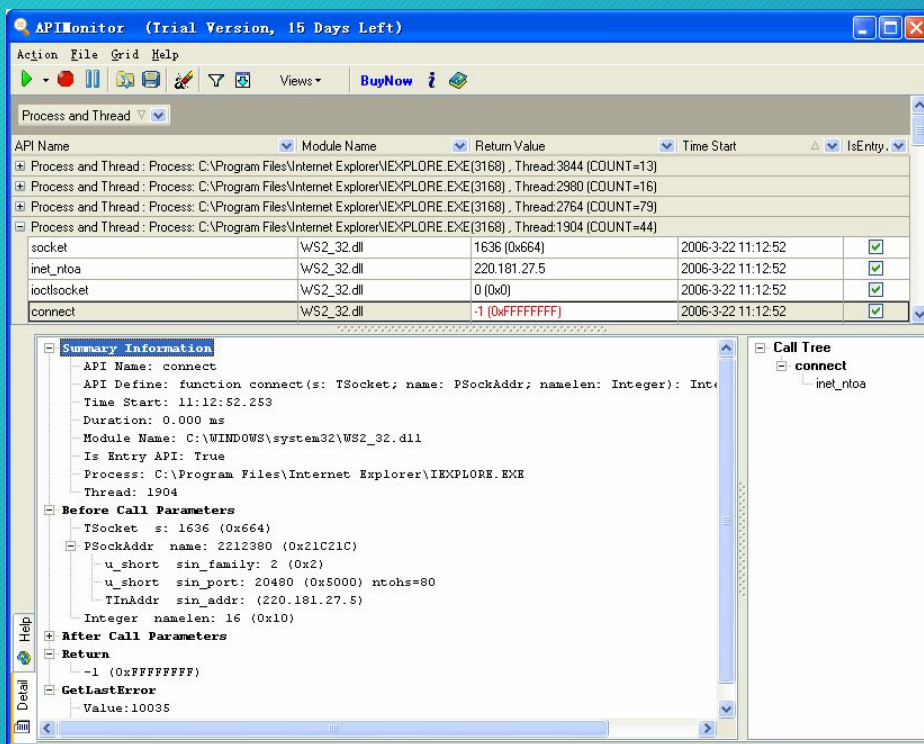
API JSON PARA JAVA

INTEGRANTES:
ROLANDO NOVOA RIOS
ELIDA PONCE ORE



API

La interfaz de programación de aplicaciones, abreviada como API del inglés: Application Programming Interface, es un conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.





{JSON}

JSON es un formato de intercambio de datos: una forma de almacenar y transferir datos. Junto con la comunicación entre el servidor y el navegador, es común ver usos como la migración de la base de datos (por ejemplo, la conversión de JSON a SQL) y la exportación de datos desde aplicaciones web propietarias.

```
{  "firstName": "John",
  "lastName": "Smith",
  "age": 25,  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": 10021
  },
  "phoneNumbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "fax",
      "number": "646 555-4567"
    }
  ]
}
```





{json:api}

- LA API DE MODELOS DE OBJETOS:

```
URL url = new URL("https://graph.facebook.com/search?q=java&type=post");
try (InputStream is = url.openStream();    JsonReader rdr = Json.createReader(is)) {
    JsonObject obj = rdr.readObject();
    JsonArray results = obj.getJsonArray("data");
    for (JsonObject result : results.getValuesAs(JsonObject.class)) {
        System.out.print(result.getJSONObject("from").getString("name"));
        System.out.print(": ");
        System.out.println(result.getString("message", ""));
        System.out.println("-----");
    }
}
```



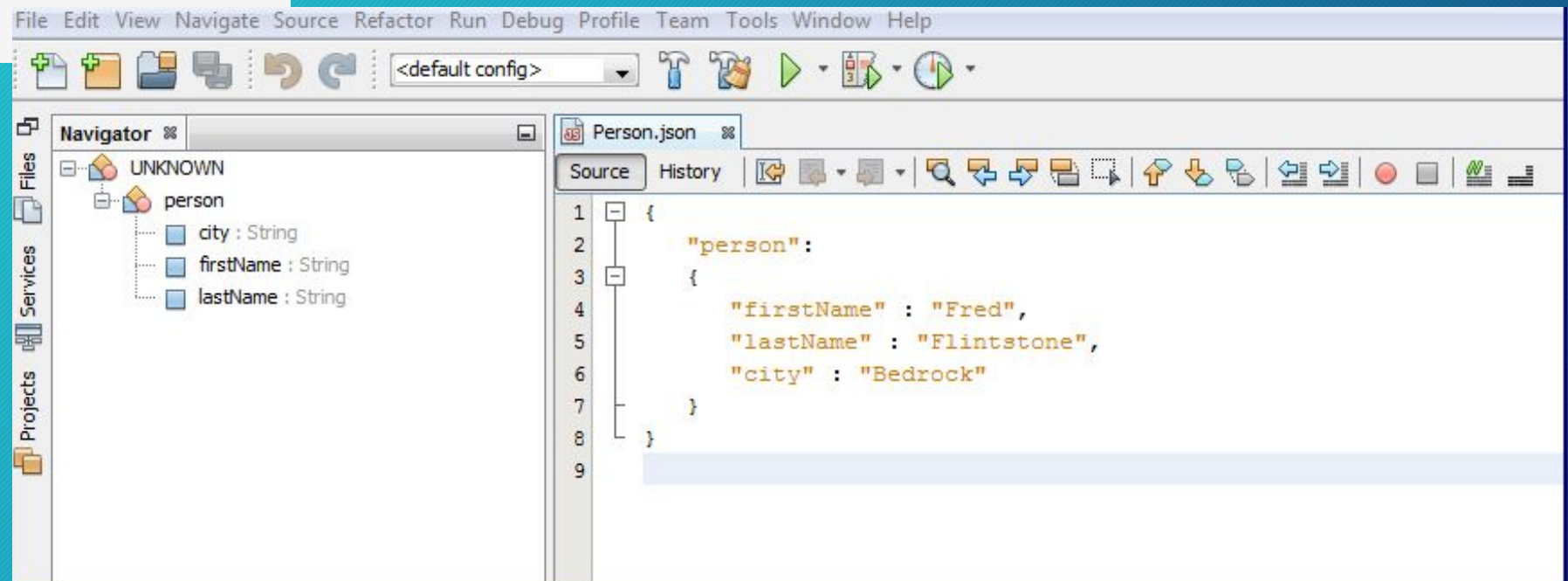

- LA API DE STREAMING :

Llevar a cabo una búsqueda de posts públicos de Facebook sobre *java*.

```
URL url = new URL("https://graph.facebook.com/search?q=java&type=post");
try (InputStream is = url.openStream()) {
    JsonParser parser = Json.createParser(is) {
    while (parser.hasNext()) {
        Event e = parser.next();
        if (e == Event.KEY_NAME) {
            switch (parser.getString()) {
                case "name":
                    parser.next();
                    System.out.print(parser.getString());
                    System.out.print(": ");
                    break;
            case "message":
                parser.next();
                System.out.println(parser.getString());
                System.out.println("-----");
                break;
            }
        }
    }
}
```


EJEMPLO EN NETBEANS

```
{  
  "person":  
  {  
    "firstName" : "Fred",  
    "lastName" : "Flintstone",  
    "city" : "Bedrock"  
  }  
}
```





GRACIAS

