

La interpretación de informes de cobertura de código es esencial para entender qué partes de tu aplicación están siendo probadas y cuáles no. Los informes de cobertura te ayudan a identificar áreas de tu código que podrían necesitar más pruebas y a mejorar la calidad de tus pruebas.

### **Porcentaje de Líneas Cubiertas**

El porcentaje de líneas cubiertas indica la proporción de líneas de código que se ejecutan durante las pruebas en comparación con el total de líneas de código.

Cómo Interpretar el Porcentaje de Líneas Cubiertas:

#### **Alto Porcentaje de Cobertura:**

**Ventaja:** Indica que una gran parte del código se está ejecutando durante las pruebas, lo que sugiere que las pruebas son exhaustivas.

**Desventaja:** No garantiza que todas las rutas o escenarios posibles estén probados. A veces, un alto porcentaje puede ser engañoso si las pruebas no cubren todos los casos de uso importantes.

#### **Bajo Porcentaje de Cobertura:**

**Ventaja:** Indica áreas donde las pruebas son insuficientes.

**Desventaja:** Un bajo porcentaje puede sugerir que hay partes significativas del código que no están siendo verificadas, lo que puede llevar a errores no detectados.

### **Identificación de Código No Probado**

El informe de cobertura te muestra qué partes del código no están siendo ejecutadas durante las pruebas. Esto se visualiza generalmente con colores o marcas en el informe.

#### **Cómo Identificar y Abordar Código No Probado:**

##### **Revisar el Informe:**

**Coloración:** Los informes de cobertura a menudo usan colores para indicar el código cubierto y no cubierto (por ejemplo, verde para cubierto y rojo para no cubierto).

**Líneas No Cubiertas:** Busca las líneas que están marcadas como no cubiertas en el informe. Estas son las áreas de tu código que no están siendo ejecutadas por las pruebas actuales.



### Analizar el Código No Cubierto:

**Casos de Uso Especiales:** Verifica si el código no cubierto incluye casos de uso especiales o caminos de código inusuales.

**Rutas de Ejecución:** Asegúrate de que todas las rutas posibles a través de tu código estén siendo probadas.

### Agregar Pruebas:

**Cobertura de Casos de Uso:** Agrega pruebas para los casos de uso y rutas que actualmente no están cubiertos.

**Pruebas de Borde y Errores:** Asegúrate de incluir pruebas para condiciones de borde y casos de error.

### Revisar el Contexto:

**Código Condicional:** Verifica si el código no cubierto está dentro de bloques condicionales que pueden no estar siendo ejecutados durante las pruebas.

**Dependencias Externas:** Asegúrate de que las pruebas cubran las interacciones con dependencias externas o servicios.

### Ejemplo de Interpretación de Informes de Cobertura

Imagina que tienes el siguiente fragmento de informe de cobertura:

File	% Stmts	% Branch	% Funcs	% Lines
src/app.js	80.00	75.00	90.00	85.00
1 files 100% of 1 covered				

### Interpretación:

**Porcentaje de Líneas Cubiertas (85.00%):** Indica que el 85% de las líneas de src/app.js están siendo ejecutadas durante las pruebas.

**Acción:** Revisa las líneas restantes (15%) que no están cubiertas.



**Porcentaje de Sentencias (80.00%):** Mide cuántas sentencias de código se ejecutan durante las pruebas.

**Acción:** Identifica las sentencias no cubiertas y agrega pruebas para esos casos.

**Porcentaje de Ramas (75.00%):** Mide cuántas ramas condicionales (if, switch, etc.) se han ejecutado en las pruebas.

**Acción:** Revisa las ramas no cubiertas y asegúrate de que las pruebas verifiquen todas las posibles rutas de ejecución.

**Porcentaje de Funciones (90.00%):** Indica cuántas funciones están siendo ejecutadas durante las pruebas.

**Acción:** Verifica las funciones no cubiertas y añade pruebas para asegurar que todas las funciones sean probadas.

### **Herramientas para Generar Informes de Cobertura**

Jest: Ofrece informes de cobertura integrados.

```
'''bash
```

```
jest --coverage
```

```
'''
```

**Istanbul/NYC:** Herramienta popular para generar informes de cobertura.

```
'''bash
```

```
nyc npm test
```

```
'''
```

Karma + Jasmine: Puedes integrar cobertura con karma-coverage para generar informes de cobertura en Karma.

```
'''bash
```

```
npm install karma-coverage --save-dev
```

```
'''
```

Configura karma.conf.js:



```
```javascript
```

Copiar código

```
coverageReporter: {  
  type: 'html',  
  dir: 'coverage/'  
}  
```
```

## Resumen

**Porcentaje de Líneas Cubiertas:** Indica qué proporción del código está siendo ejecutada durante las pruebas.

**Identificación de Código No Probado:** Utiliza los informes de cobertura para encontrar y agregar pruebas para el código no cubierto.

**Herramientas:** Usa herramientas como Jest, Istanbul/NYC, y Karma para generar y revisar informes de cobertura.

Asegurarse de que todas las partes importantes de tu código estén cubiertas por pruebas ayuda a mejorar la calidad y fiabilidad del software.