

Prompt (input):

@SafeVarargs

```
public static <T> List<T> buildListOfAtLeastTwo(T first, T second, T... more) {  
    if (first == null) {  
        throw new IllegalArgumentException("First example is null.");  
    }  
    if (second == null) {  
        throw new IllegalArgumentException("Second example is null.");  
    }  
  
    List<T> result = new ArrayList<>();  
    result.add(first);  
    result.add(second);  
    addArrayElementsToList(result, more);  
  
    return result;  
}
```

Please find some info about the location of the function in the repo.

Repo: jqno/equalsverifier

Path: src/main/java/nl/jqno/equalsverifier/internal/util/ListBuilders.java

Function_name: ListBuilders.buildListOfAtLeastTwo

We categorized the identifiers into different classes. Please find the information below.

Function name: ListBuilders.buildListOfAtLeastTwo

Parameters of the function: ['first', 'second']

Identifier to be returned: ['result']

Method Invocation: ['result', 'add', 'addArrayElementsToList']

Method Arguments: ['first', 'second', 'result', 'more']

Variable declaration: ['more', 'result']

Please find the dataflow of the function. We present the source and list of target indices.

T-[5] [9, 13, 16, 19, 58]

first-[14] [28, 73]

second-[17] [43, 80]

IllegalArgumentException-[35] [50]

List-[7] [56]

result-[60] [69, 76, 85, 91]

Write down the original comment written by the developer.

Comment: Builds a list with at least two examples .

```
public Observable<ServiceResponse<OrchestratorVersionProfileListResultInner>>  
listOrchestratorsWithServiceResponseAsync(String location) {  
    if (this.client.subscriptionId() == null) {
```

```

        throw new IllegalArgumentException("Parameter this.client.subscriptionId() is required
and cannot be null.");
    }
    if (location == null) {
        throw new IllegalArgumentException("Parameter location is required and cannot be
null.");
    }
    if (this.client.apiVersion() == null) {
        throw new IllegalArgumentException("Parameter this.client.apiVersion() is required and
cannot be null.");
    }
    final String resourceType = null;
    return service.listOrchestrators(this.client.subscriptionId(), location, this.client.apiVersion(),
resourceType, this.client.acceptLanguage(), this.client.userAgent())
        .flatMap(new Func1<Response<ResponseBody>,
Observable<ServiceResponse<OrchestratorVersionProfileListResultInner>>>() {
            @Override
            public Observable<ServiceResponse<OrchestratorVersionProfileListResultInner>>
call(Response<ResponseBody> response) {
                try {
                    ServiceResponse<OrchestratorVersionProfileListResultInner> clientResponse =
listOrchestratorsDelegate(response);
                    return Observable.just(clientResponse);
                } catch (Throwable t) {
                    return Observable.error(t);
                }
            }
        });
}

```

Please find some info about the location of the function in the repo.

Repo: Azure/azure-sdk-for-java

Path:

containerservice/resource-manager/v2017_09_30/src/main/java/com/microsoft/azure/managem
ent/containerservice/v2017_09_30/implementation/ContainerServicesInner.java

Function_name: ContainerServicesInner.listOrchestratorsWithServiceResponseAsync

We categorized the identifiers into different classes. Please find the information below.

Function name: ContainerServicesInner.listOrchestratorsWithServiceResponseAsync

Parameters of the function: ['location', 'response']

Method Invocation: ['subscriptionId', 'apiVersion', 'service', 'listOrchestrators', 'acceptLanguage',
'userAgent', 'flatMap', 'listOrchestratorsDelegate', 'Observable', 'just', 'error']

Method Arguments: ['location', 'resourceType', 'response', 'clientResponse', 't']

Variable declaration: ['resourceType', 'clientResponse']

Identifier to access field/array: ['client']

Please find the dataflow of the function. We present the source and list of target indices.

location-[10] [36, 89]
IllegalArgumentException-[28] [43, 64]
String-[9] [71]
resourceType-[72] [99]
Observable-[1] [128, 140, 171]
ServiceResponse-[3] [130, 142, 158]
OrchestratorVersionProfileListResultInner-[5] [132, 144, 160]
public-[0] [139]
Response-[123] [149]
ResponseBody-[125] [151]
t-[167] [175]

Write down the original comment written by the developer.

Comment: Gets a list of supported orchestrators in the specified subscription . Gets a list of supported orchestrators in the specified subscription . The operation returns properties of each orchestrator including version and available upgrades .

```
public static byte[] calculateBlockHash(HFClient client, long blockNumber, byte[] previousHash,
byte[] dataHash) throws IOException, IllegalArgumentException {
```

```
    if (previousHash == null) {
        throw new IllegalArgumentException("previousHash parameter is null.");
    }
    if (dataHash == null) {
        throw new IllegalArgumentException("dataHash parameter is null.");
    }
    if (null == client) {
        throw new IllegalArgumentException("client parameter is null.");
    }

    CryptoSuite cryptoSuite = client.getCryptoSuite();
    if (null == cryptoSuite) {
        throw new IllegalArgumentException("Client crypto suite has not been set.");
    }

    ByteArrayOutputStream s = new ByteArrayOutputStream();
    DERSequenceGenerator seq = new DERSequenceGenerator(s);
    seq.addObject(new ASN1Integer(blockNumber));
    seq.addObject(new DEROctetString(previousHash));
    seq.addObject(new DEROctetString(dataHash));
    seq.close();
    return cryptoSuite.hash(s.toByteArray());

}
```

Please find some info about the location of the function in the repo.

Repo: hyperledger/fabric-sdk-java

Path: src/main/java/org/hyperledger/fabric/sdk/SDKUtils.java

Function_name: SDKUtils.calculateBlockHash

We categorized the identifiers into different classes. Please find the information below.

Function name: SDKUtils.calculateBlockHash

Parameters of the function: ['client', 'blockNumber', 'previousHash', 'dataHash']

Method Invocation: ['client', 'getCryptoSuite', 'seq', 'addObject', 'close', 'cryptoSuite', 'hash', 's', 'toByteArray']

Method Arguments: ['s', 'blockNumber', 'previousHash', 'dataHash']

Variable declaration: ['cryptoSuite', 's', 'seq']

Please find the dataflow of the function. We present the source and list of target indices.

byte-[2] [13, 18]

previousHash-[16] [30, 132]

InvalidArgumentException-[26] [37, 52, 67, 91]

dataHash-[21] [45, 143]

if-[43] [58]

client-[8] [62, 76]

throw-[50] [65, 89]

new-[51] [66, 90]

cryptoSuite-[74] [86, 154]

ByteArrayOutputStream-[97] [101]

DERSequenceGenerator-[105] [109]

s-[98] [111, 158]

seq-[106] [114, 125, 136, 147]

blockNumber-[11] [121]

DEROctetString-[130] [141]

Write down the original comment written by the developer.

Comment: used asn1 and get hash

```
public void addPacketInterceptor(PacketInterceptor packetInterceptor,
    PacketFilter packetFilter) {
    if (packetInterceptor == null) {
        throw new NullPointerException("Packet interceptor is null.");
    }
    interceptors.put(packetInterceptor, new InterceptorWrapper(
        packetInterceptor, packetFilter));
}
```

Please find some info about the location of the function in the repo.

Repo: abmargb/jamppa

Path: src/main/java/org/jivesoftware/smack/Connection.java

Function_name: Connection.addPacketInterceptor

Please find the dataflow of the function. We present the source and list of target indices.

packetInterceptor-[5] [13, 30, 35]

packetFilter-[8] [37]

We categorized the identifiers into different classes. Please find the information below.

Function name: Connection.addPacketInterceptor

Parameters of the function: ['packetInterceptor', 'packetFilter']

Method Invocation: ['interceptors', 'put']

Method Arguments: ['packetInterceptor', 'packetFilter']

Write down the original comment written by the developer.

Comment:

Model output:

Adds a packet interceptor that will be invoked every time a packet is about to be sent to the server. The interceptor has the option to modify the packet that is about to be sent. A packet filter determines which packets will be delivered to the interceptor.