|  |  |  |  |
| --- | --- | --- | --- |
| ID | Title and post link | ChatGPT Answer | Overall Labels |
| 75251151 | React Native: Execution failed for task ':app:checkDebugDuplicateClasses'  <https://stackoverflow.com/questions/75251151/react-native-execution-failed-for-task-appcheckdebugduplicateclasses> | This error usually occurs when there are conflicting versions of Kotlin libraries in your React Native project. It seems like the issue is specifically related to the `react-native-inappbrowser-reborn` library.    Here are a few steps you can try to resolve this issue:    1. Clean the project: Run the following commands in your project's root directory:  ```bash  cd android  ./gradlew clean *[Incorrect (Concept), Inconsistent (Concept), Not Concise (Irrelevant)]*  ```    2. Check dependencies: Open the `android/app/build.gradle` file and make sure that you have only one version of the Kotlin library dependencies. In your case, it should be `"org.jetbrains.kotlin:kotlin-stdlib-jdk8:1.7.20"` for both `implementation` and `api` configurations.    You can remove the `kotlinVersion = "1.8.0"` line from `android/build.gradle` as it seems unnecessary. *[Incorrect (Factual), Inconsistent (Factual)]*    3. Check React Native libraries: Make sure all your React Native libraries are compatible with each other and with the version of React Native you are using. Some libraries may have their own Kotlin dependencies that could conflict with the ones used by React Native.    If you find any conflicting dependencies, try updating the libraries to their latest versions or find alternative libraries that are compatible with your current setup.    4. Invalidate caches and restart: In Android Studio, go to `File -> Invalidate Caches / Restart` and select "Invalidate and Restart." This action will clear any cached build files and force a fresh build. *[Incorrect (Concept), Inconsistent (Concept), Not Concise (Irrelevant)]*    5. Upgrade React Native: If none of the above steps work, you can try upgrading React Native to the latest stable version. This may resolve compatibility issues between libraries and provide a more recent version of Kotlin dependencies. *[Incorrect (Concept), Inconsistent (Concept), Not Concise (Irrelevant)]*    However, keep in mind that upgrading React Native may require additional adjustments in your project, so make sure to thoroughly test your app after the upgrade. *[Not Concise (Irrelevant)]*    If you still encounter the same error after trying these steps, it's possible that there is an issue with the `react-native-inappbrowser-reborn` library itself. In that case, you can consider reaching out to the library's maintainers for assistance or look for alternative libraries that provide similar functionality.    Remember to always backup your project before making any major changes, and carefully review the documentation and release notes of any libraries or tools you are using. *[Not Concise (Irrelevant)]* | Not Comprehensive  Usefulness (1)  -------------------  Incorrect  Inconsistent  Not Concise |
| 75252277 | Weird Python Selenium error message when asked to find an element  <https://stackoverflow.com/questions/75252277/weird-python-selenium-error-message-when-asked-to-find-an-element> | The error message you're seeing is a combination of two different issues. *[Incorrect (Concept), Inconsistent (Concept)]* Let's break it down: *[Inconsistent (Number of Solutions)]*    1. USB-related Error Messages:  ```  [23248:27252:0126/162232.236:ERROR:device\_event\_log\_impl.cc(215)] [16:22:32.235] USB: usb\_service\_win.cc:415 Could not read device interface GUIDs: The system cannot find the file specified. (0x2)  [23248:27252:0126/162232.237:ERROR:device\_event\_log\_impl.cc(215)] [16:22:32.238] USB: usb\_device\_handle\_win.cc:1046 Failed to read descriptor from node connection: A device attached to the system is not functioning. (0x1F)  ```    These error messages are related to USB device handling. They are not directly related to your code or Selenium. They indicate that the system is unable to read device interface GUIDs and descriptors from a USB connection. These messages are often seen in the Chrome browser's debug log and can be safely ignored for your scenario. *[Incorrect (Concept), Inconsistent (Concept), Not Concise (Irrelevant)]*  2. NoSuchElementException:  ```  selenium.common.exceptions.NoSuchElementException: Message: no such element: Unable to locate element: {"method":"css selector","selector":".start is-armed"}  ```    This exception occurs when Selenium is unable to find an element on the web page matching the specified selector. In your case, the selector is `.start is-armed` using CSS class names.    The issue is likely with the selector you are using. The selector `.start is-armed` is looking for an element with both the classes `start` and `is-armed`, but it seems they are separate classes applied to different elements.    To fix this, you can modify the selector to use a different CSS selector strategy, such as combining the classes using a single dot (indicating multiple classes on the same element) or using a different attribute or element type to identify the desired element.    For example, if the element has both classes `start` and `is-armed`, you can try using:  ```python  startButton = driver.find\_element(By.CSS\_SELECTOR, ".start.is-armed")  ```    Alternatively, you can use other selectors based on the HTML structure or attributes of the element you want to locate. You can inspect the HTML of the page to identify unique attributes or parent elements that can be used to target the desired element more accurately. *[ Not Concise (Irrelevant)]*    Remember to ensure that the element you are trying to locate is present in the page's DOM at the time the `find\_element` method is called. If the element is dynamically loaded or rendered after some action or delay, you may need to add appropriate waits in your code to ensure the element is available before attempting to locate it.    By addressing the CSS selector issue and making sure the element is present, you should be able to resolve the `NoSuchElementException` and successfully locate the desired element with Selenium. *[Not Concise (Excess)]* | Not Comprehensive  Usefulness (4)  -------------------  Partially Correct  Partially Consistent  Not Concise |
| 75256018 | C# triple double quotes (three double quotes)  <https://stackoverflow.com/questions/75256018/c-sharp-triple-double-quotes-three-double-quotes> | In C#, triple double quotes (`"""`) are used to represent a verbatim string literal. Verbatim string literals allow you to include line breaks and escape sequences without using additional escape characters. They are particularly useful when dealing with multi-line text, such as long paragraphs, SQL queries, XML, or JSON.    Here's an example:    ```csharp  string text = @"  This is a multi-line  string literal using  verbatim syntax.  "; *[Incorrect Code (Wrong Syntax), Inconsistent (Code)]*  ```    In the above example, the string `text` will contain the entire multi-line text without any additional escape characters. The leading `@` symbol before the opening double quotes indicates that it's a verbatim string literal. *[Incorrect (Factual), Inconsistent (Factual)]*    The alternative approach you mentioned, using `@"..."`, is called a verbatim interpolated string. It allows you to include interpolation expressions (`$"{expression}"`) within the string while still preserving the verbatim nature of the string.    Here's an example of a verbatim interpolated string:    ```csharp  string name = "John";  int age = 30;  string info = $@"Name: {name}  Age: {age}";  ``` *[Incorrect Code (Wrong Syntax), Inconsistent (Code)]*    In the above example, the string `info` will contain the interpolated values of `name` and `age` without any additional escape characters. The `@` symbol before the opening double quotes makes it a verbatim string literal, and the `$` symbol enables string interpolation. *[Incorrect (Factual), Inconsistent (Factual)]*  The choice between using triple double quotes (`"""`) and verbatim interpolated strings (`$@"..."`) depends on your specific use case. If you need to include multi-line text without any interpolation, triple double quotes are a simple and convenient option. However, if you need to include interpolated values within the multi-line text, verbatim interpolated strings provide the necessary functionality. *[Not Concise (Excess)]* | Fully Comprehensive  Usefulness (2)  -------------------  Incorrect  Inconsistent  Partially Concise |
| 75258289 | Correct the classpath of your application so that it contains compatible versions of the classes Log4J2LoggingSystem and PropertiesUtil  <https://stackoverflow.com/questions/75258289/correct-the-classpath-of-your-application-so-that-it-contains-compatible-version> | The error message suggests that there is a mismatch between the versions of Log4j2 and PropertiesUtil classes being used in your application. To resolve this issue, you need to ensure that the classpath of your application contains compatible versions of these classes. Here's how you can do it: *[Inconsistent (Number of Solutions)]*    1. Update your Maven dependencies to use compatible versions of Log4j2 and SLF4J. In your pom.xml file, make the following changes:    ```xml  <!-- Update log4j2-slf4j-impl version -->  <dependency>  <groupId>org.apache.logging.log4j</groupId>  <artifactId>log4j-slf4j-impl</artifactId>  <version>2.18.0</version> <!-- Replace with a compatible version -->  </dependency>    <!-- Update slf4j-api version -->  <dependency>  <groupId>org.slf4j</groupId>  <artifactId>slf4j-api</artifactId>  <version>1.7.32</version> <!-- Replace with a compatible version -->  </dependency> *[Incorrect Code (Wrong API/Library/Function Usage), Inconsistent (Code)]*  ```    2. Make sure you don't have conflicting dependencies in your classpath that may bring in incompatible versions of Log4j2 or SLF4J. You can use the Maven Dependency Plugin to analyze your dependency tree and check for conflicts. Run the following command in your project directory:    ```  mvn dependency:tree  ```    Look for any dependencies that may be causing conflicts or bringing in older versions of Log4j2 or SLF4J. If you find any conflicting dependencies, you can exclude them or try to find alternative dependencies that are compatible with your Spring Boot version. *[Not Concise (Redundant)]*    3. Clean your project and rebuild it to ensure that the updated dependencies are properly resolved and included in your classpath.    4. If you're using an IDE like IntelliJ IDEA or Eclipse, refresh your project to ensure that the IDE's internal dependency resolution is updated. *[Incorrect (Concept), Inconsistent (Concept),Not Concise (Irrelevant)]*    By correcting the classpath and ensuring that you have compatible versions of Log4j2 and PropertiesUtil, you should be able to resolve the issue related to SLF4J and Log4j2 compatibility. *[Not Concise (Redundant)]* | Fully Comprehensive  Usefulness (3)  -------------------  Incorrect  Inconsistent  Not Concise |
| 75261905 | How should I resolve this Angular node package module dependency conflict?  <https://stackoverflow.com/questions/75261905/how-should-i-resolve-this-angular-node-package-module-dependency-conflict> | In this situation, it's generally not recommended to use the `--force` or `--legacy-peer-deps` flags unless you have exhausted all other options, as they can lead to potential compatibility issues and broken dependencies.    To resolve the Angular node package module dependency conflict, you can try the following steps:    1. Update the `ngx-gallery-9` package: Check if there is a newer version of `ngx-gallery-9` available that has an updated peer dependency on `tslib`. You can check the package's repository or the npm registry to find the latest version. If a newer version is available, update the package in your project's `package.json` file to use that version.    2. Resolve conflicting dependencies: If updating `ngx-gallery-9` doesn't resolve the conflict, you can try updating other packages that depend on `tslib` as well. Look for packages in your project that have `tslib` as a dependency and check if there are newer versions available that align with the required version of `ngx-gallery-9`. Update those packages accordingly.    3. Verify compatibility: After updating the packages, verify that the updated versions are compatible with the other dependencies in your project. Run your application and conduct thorough testing to ensure everything works as expected.    4. Report the issue: If you are unable to resolve the conflict by updating the packages, consider reporting the issue to the maintainers of `ngx-gallery-9` and any other relevant packages. They may be able to provide guidance or release updated versions with resolved dependencies.    Remember to always backup your project and its dependencies before making any significant changes. This ensures you can revert back to a stable state if needed. *[Not Concise (Excess)]* | Fully Comprehensive  Usefulness (5)  -------------------  Correct  Consistent  Not Concise |
| 75268665 | Micrometer tracing does not include the trace ID in the log  <https://stackoverflow.com/questions/75268665/micrometer-tracing-does-not-include-the-trace-id-in-the-log> | To include the Trace ID in your application logs when using the Micrometer tracing library, you need to configure the logging framework to include the Trace ID as a variable in the log pattern. In your case, you are using Log4j 2 as the logging framework. *[Incorrect (Concept), Inconsistent (Concept)]*    Here's what you need to do:    1. Update the `logging.pattern.level` property in your `application.yml` file to include the `%X{traceId:-}` variable in the log pattern. Make sure to place it within the square brackets alongside other variables:    ```yaml  logging:  pattern:  level: "%5p [${spring.application.name:},%X{traceId:-},%X{spanId:-}]"  ```    This configuration uses `%X{traceId:-}` to include the Trace ID in the log pattern. If the Trace ID is not available, it will be represented as a hyphen (`-`). *[Incorrect (Concept), Inconsistent (Concept), Incorrect (Terminology)]*    2. Ensure that you have the necessary dependencies in your project. From your `pom.xml`, it seems that you have already included the required dependencies for Micrometer and Log4j 2. Make sure they are up to date.    3. Verify the log output. Once you start your application, the logs should now include the Trace ID alongside other log information. *[Incorrect (Concept), Inconsistent (Concept), Incorrect (Terminology)]*    Please note that the actual log output may vary depending on your Log4j 2 configuration and the log appender you are using. Make sure you have the appropriate Log4j 2 configuration file (e.g., `log4j2.xml` or `log4j2.properties`) in your project and it is properly configured to output logs with the desired pattern.    If you continue to face issues, make sure to check your Log4j 2 configuration and consult the Log4j 2 documentation for further guidance on configuring log patterns and variables. *[Not Concise (Excess)]* | Fully Comprehensive  Usefulness (1)  -------------------  Incorrect  Inconsistent  Not Concise |
| 75270837 | Cannot read properties of undefined (reading '$refs') vue js  <https://stackoverflow.com/questions/75270837/cannot-read-properties-of-undefined-reading-refs-vue-js> | In Vue.js, the `$refs` property allows you to access references to child components, DOM elements, or child elements within a component's template. *[Not Concise (Excess)]* However, when accessing `$refs` directly in the script section of your component, you need to make sure that the DOM element you are trying to access is available.    In your case, you are trying to access the `input` element using `$refs.input` outside of any lifecycle hook, which can lead to the error you're encountering (`Cannot read properties of undefined (reading '$refs')`). To resolve this issue, you can indeed use the `mounted` lifecycle hook to ensure that the component's template has been rendered and the element you want to reference is available.    Here's an example of how you can modify your code to use the `mounted` hook:    ```vue  <template>  <div class="input-wrapper">  <input type="text" id="phone" placeholder="(555) 555-5555" ref="input" />  </div>  </template>    <script>  export default {  mounted() {  this.$refs.input.addEventListener('input', () => {  // Your code here  });  },  };  </script>  ```    By moving your code inside the `mounted` hook, you ensure that the component has been mounted and the `$refs.input` is available for accessing.    Alternatively, you can also use the `updated` hook if you need to perform this logic whenever the component or the `input` element is updated:    ```vue  <script>  export default {  updated() {  this.$refs.input.addEventListener('input', () => {  // Your code here  });  },  };  </script>  ```    In this case, the logic will be triggered whenever the component or the `input` element is updated. Choose the appropriate hook based on your specific requirements. | Not Comprehensive  Usefulness (5)  -------------------  Correct  Consistent  Partially Concise |
| 75273409 | compiled with an incompatible version of Kotlin. The binary version of its metadata is 1.8.0, expected version is 1.6.0  <https://stackoverflow.com/questions/75273409/compiled-with-an-incompatible-version-of-kotlin-the-binary-version-of-its-metad> | The error message suggests that there is an incompatibility between the Kotlin version used to compile one of the modules and the version expected by your project. To resolve this issue, you can try the following steps:    1. Update the Kotlin version in your project:  - In your project-level `build.gradle` file, make sure you are using the latest Kotlin version. Update the `ext.kotlin\_version` property to `'1.8.0'` instead of `'1.6.10'`.    2. Clean and rebuild the project:  - Run a clean build of your project. In Android Studio, go to `Build` > `Clean Project` to clean the project.  - After the clean process completes, rebuild your project by going to `Build` > `Rebuild Project`.    If the issue persists after following the above steps, you can try the following additional suggestions:    3. Check for any conflicting Kotlin versions:  - Ensure that there are no dependencies or plugins in your project that are explicitly using an older version of Kotlin. Check your project's dependencies and plugins to ensure they are compatible with Kotlin version 1.8.0.    4. Invalidate caches and restart Android Studio:  - Go to `File` > `Invalidate Caches / Restart` in Android Studio to clear any cached data that might be causing conflicts.    5. Update other Kotlin-related dependencies:  - Check if any of your dependencies have updates available for their Kotlin-related libraries. Update them to the latest compatible versions. *[Incorrect (Concept), Inconsistent (Concept), Not Concise (Irrelevant)]*    If none of the above solutions work, you may need to provide more information about your project setup and any additional error messages for further assistance. *[Not Concise (Excess)]* | Not Comprehensive  Usefulness (3)  -------------------  Partially Incorrect  Partially Inconsistent  Not Concise |
| 75278610 | TypeError: Cannot read properties of undefined (reading 'id') in Expressjs  <https://stackoverflow.com/questions/75278610/typeerror-cannot-read-properties-of-undefined-reading-id-in-expressjs> | The error message suggests that you are trying to access the `id` property of an object, but the object is `undefined`. In your code, the issue seems to be with the line where you're trying to access `user.id` in the `jwt.sign()` function.    To fix the issue, you should check if the `user` object is defined before accessing its properties. You can modify your code as follows: *[Inconsistent (Number of Solutions)]*    ```typescript  async function Register(req: Request, res: Response): Promise<Response> {  try {  const { id, first\_name, last\_name, email, password, token } = req.body;  await Users.find(email);  const encrypt: typeof Users = await bcrypt.hash(password, 10);  const user: typeof Users = await Users.create({ first\_name, last\_name, email: email.toLowerCase(), password: encrypt });  const userId = user?.id?.toString(); // Check if user and user.id are defined  const Token: typeof Users = await jwt.sign({ user\_id: userId, email: user?.email }, process.env.TOKEN\_SECRET);  user.token = Token;  return res.status(200).json(user);  } catch (error) {  console.error('User could not be registered', error);  // Handle the error and send an appropriate response  return res.status(500).json({ error: 'Internal Server Error' });  } *[Incorrect Code (Incomplete), Inconsistent (Code)]*  }  ```    In the modified code, we use the optional chaining operator (`?.`) to check if `user` and `user.id` are defined before accessing the `id` property. If either `user` or `user.id` is `undefined`, the `userId` variable will be assigned `undefined`. This helps prevent the `TypeError` from occurring.    Additionally, I added an error handling block to catch any errors that might occur during the registration process. In case of an error, an appropriate response with a 500 status code is sent back to the client.    Make sure to update any other parts of your code that rely on the `id` property of the `user` object to handle the possibility of it being `undefined` as well. *[Not Concise (Excess)]* | Not Comprehensive  Usefulness (4)  -------------------  Partially Incorrect  Partially Inconsistent  Partially Concise |
| 75284194 | pandasql.sqldf(); ObjectNotExecutableError: Not an executable object: 'SELECT \* FROM df '  <https://stackoverflow.com/questions/75284194/pandasql-sqldf-objectnotexecutableerror-not-an-executable-object-select> | The issue you're facing is related to the usage of `pandasql` library. *[Incorrect (Concept), Inconsistent (Concept)]*It seems that the library is unable to execute the SQL query on the DataFrame object.    To resolve this issue, you can use the `sqldf()` function from the `pandasql` library in a slightly different way. Instead of passing the DataFrame object directly in the query, you need to specify the DataFrame as a table within the query. *[Incorrect (Concept), Inconsistent (Concept)]*    Here's an updated version of your code that should work:    ```python  import pandas as pd  import pandasql as ps  import numpy as np    df = pd.DataFrame([[1234, 'Customer A', '123 Street', np.nan],  [1234, 'Customer A', np.nan, '333 Street'],  [1233, 'Customer B', '444 Street', '333 Street'],  [1233, 'Customer B', '444 Street', '666 Street']],  columns=['ID', 'Customer', 'Billing Address', 'Shipping Address'])    # Register the DataFrame as a table  ps.sqldf("SELECT \* FROM df", locals()) *[Incorrect Code (Wrong API/Library/Function Usage), Inconsistent (Concept)]*  ```    In the updated code, the DataFrame `df` is registered as a table using the `locals()` function, which provides a dictionary of local variables. This allows `pandasql` to recognize `df` as a table within the SQL query. *[Not Concise (Redundant)]*    By making this change, the SQL query should execute properly on the DataFrame, and you should get the expected result. *[Incorrect (Concept), Inconsistent (Concept), Not Concise (Redundant)]* | Fully Comprehensive  Usefulness (1)  -------------------  Incorrect  Inconsistent  Partially Concise |
| 75284995 | Spring Reactor: adding delay but in an NON blocking way  <https://stackoverflow.com/questions/75284995/spring-reactor-adding-delay-but-in-an-non-blocking-way> | To simulate a delay in a non-blocking way using Spring Reactor, you can use the `Mono.delay(Duration)` operator. This operator creates a Mono that emits a single item (or completion signal) after the specified delay duration.    Here's an example of how you can modify your code to introduce a non-blocking delay:    ```java  @Test  public void simulateLengthyProcessingOperationReactor() {  Flux.range(1, 5000)  .concatMap(a -> simulateLengthyProcessingOperation(a))  .subscribe(System.out::println);  }    public Mono<String> simulateLengthyProcessingOperation(Integer input) {  return simulateDelay\_NON\_blocking()  .thenReturn(String.format("[%d] on thread [%s] at time [%s]", input, Thread.currentThread().getName(), new Date()));  }    public Mono<Void> simulateDelay\_NON\_blocking() {  return Mono.delay(Duration.ofSeconds(4))  .then();  }  ```    In the modified code, `simulateLengthyProcessingOperation()` returns a `Mono<String>` instead of a `String`, indicating that the method is asynchronous. Within `simulateLengthyProcessingOperation()`, `simulateDelay\_NON\_blocking()` returns a `Mono<Void>` that represents the non-blocking delay. By using `.thenReturn()` after the delay, we transform the `Mono<Void>` into a `Mono<String>` with the desired result.    The `concatMap()` operator is used instead of `map()` to preserve the order of the emitted values. This ensures that the delay for each element is applied one after another.    By using `Mono.delay(Duration)` and leveraging the reactive operators, you can introduce a delay in a non-blocking way and simulate a lengthy process within the reactive pipeline. *[Not Concise (Redundant)]* | Fully Comprehensive  Usefulness (5)  -------------------  Correct  Consistent  Not Concise |
| 75286648 | multiple profiles in one profiles.yml is possible?  <https://stackoverflow.com/questions/75286648/multiple-profiles-in-one-profiles-yml-is-possible> | Yes, it is possible to have multiple profiles in one `profiles.yml` file in dbt. Each profile can have its own set of outputs and configurations.    In your example, you have defined two profiles: `dbt\_project` and `azure\_blob`. However, there seems to be a formatting issue in your `profiles.yml` file. The `azure\_blob` profile is not properly nested under the `outputs` section of the `dbt\_project` profile.    Here's the corrected version of your `profiles.yml` file:    ```yaml  default: dbt\_project    dbt\_project:  target: dev  outputs:  dev:  type: synapse  driver: 'ODBC Driver 17 for SQL Server'  server: XXXXXXX  database: XXXXXXX  port: 1433  schema: XXXXXXX  user: XXXXXXX  password: XXXXXXX    azure\_blob:  target: dev  outputs:  dev:  type: azure\_blob  account\_name: XXXXXXX  account\_key: XXXXXXX  container: data-platform-archive  prefix: abc/FGr1/fox/  ``` *[Incorrect Code (Wrong API/Library/Function Usage), Inconsistent (Code),*    Note that I have corrected the indentation to properly nest the `azure\_blob` profile under the `outputs` section of `dbt\_project`. *[Incorrect (Concept), Inconsistent (Concept)]*    Regarding the error you're encountering when trying to read the file from Azure Blob Storage, make sure you have configured the necessary dependencies and plugins for using Azure Blob Storage as a target in dbt. Additionally, ensure that the container and file paths are correctly specified in your model file. *[Incorrect (Concept), Inconsistent (Concept), Not Concise (Excess)]*    Double-check the `connection` parameter in your model file to make sure it matches the target profile name (`azure\_blob`). Also, verify that the path and file name (`'data-platform-archive/abc/FGr1/fox/', 'abc.parquet'`) are correct and accessible in Azure Blob Storage. *[Incorrect (Concept), Inconsistent (Concept), Not Concise (Excess)]*    If the issue persists, please provide more details about the specific error message or any additional error logs for further assistance. | Partially Comprehensive  Usefulness (1)  -------------------  Incorrect  Inconsistent  Not Concise |
| 75287523 | Spring Boot 3 webflux is not compatible with springdoc openapi 2  <https://stackoverflow.com/questions/75287523/spring-boot-3-webflux-is-not-compatible-with-springdoc-openapi-2> | The issue you're facing is related to compatibility between Spring Boot 3's WebFlux and Springdoc OpenAPI 2.    Unfortunately, Springdoc OpenAPI 2 is not compatible with Spring Boot 3's WebFlux due to changes in the underlying infrastructure and dependencies. Springdoc OpenAPI 2 is designed to work with Spring Boot 2.x versions that use Spring MVC as the default web framework.    To generate API documentation for your Spring Boot 3 WebFlux application, you can use Springdoc OpenAPI 3, which is compatible with WebFlux. Here's how you can add it to your build.gradle:    ```groovy  implementation group: 'org.springdoc', name: 'springdoc-openapi-webflux-ui', version: '1.6.3'  ```    With this dependency, you should be able to generate and access the Swagger UI for your WebFlux application. You can access it at the `/swagger-ui.html` endpoint. *[Not Concise (Redundant)]*    Note that the `springdoc-openapi-webflux-ui` library uses the OpenAPI 3 specification, which is the latest version of the OpenAPI specification. You may need to update your existing Swagger annotations and configuration to match the OpenAPI 3 syntax.    Additionally, make sure you have the necessary configuration in your application to enable Springdoc OpenAPI. You can refer to the Springdoc OpenAPI documentation for more details on configuration options and usage examples. *[Not Concise (Irrelevant)]*    Remember to clean and rebuild your project after making these changes to ensure the new dependencies are properly resolved.    I hope this helps! Let me know if you have any further questions. *[Not Concise (Excess)]* | Not Comprehensive  Usefulness (3)  -------------------  Correct  Partially Inconsistent  Not Concise |
| 75294174 | Vite pwa plugin not working in development environment for react apps  <https://stackoverflow.com/questions/75294174/vite-pwa-plugin-not-working-in-development-environment-for-react-apps> | In Vite, the PWA plugin expects the service worker file to be located in the public directory by default. However, if you want to use a custom path for the service worker in the development environment, you can configure Vite to serve the service worker from the src directory. *[Incorrect (Concept), Inconsistent (Concept)]*  Here's how you can achieve that: *[Inconsistent (Number of Solutions)]*    1. Create a new file named `vite.config.js` in the root directory of your project (if it doesn't exist already).    2. Add the following code to the `vite.config.js` file:    ```javascript  import { defineConfig } from 'vite';  import VitePWA from 'vite-plugin-pwa';    export default defineConfig({  plugins: [  VitePWA({  srcDir: 'src',  filename: 'sw.js',  base: '/',  devServer: {  writeToDisk: true,  },  }),  ],  });  ```    This configuration tells Vite to serve the service worker from the `src` directory instead of the `public` directory in the development environment.    3. Start your development server using the `vite` command.    Now, when you run the development server, Vite will serve the service worker from the `src` directory, and you should no longer see the 404 error. *[Incorrect (Concept), Inconsistent (Concept)]*    Please note that this configuration is specific to the Vite PWA plugin (`vite-plugin-pwa`). If you are using a different PWA plugin, the configuration might be slightly different. Make sure to refer to the documentation of the specific plugin you are using for accurate configuration details. *[Not Concise (Excess)]*    I hope this helps! Let me know if you have any further questions. | Not Comprehensive  Usefulness (1)  -------------------  Incorrect  Inconsistent  Not Concise |
| 75295783 | Can't access attributes on a list of objects in terraform  <https://stackoverflow.com/questions/75295783/cant-access-attributes-on-a-list-of-objects-in-terraform> | The error message you're seeing is because the `for\_each` expression in your dynamic block is expecting a map, but `var.policy` is defined as a map of lists. To resolve this issue, you need to update the type definition of the `policy` variable. *[Incorrect (Concept), Inconsistent (Concept)]*    Here's the updated variable definition:    ```terraform  variable "policy" {  description = "The policy to be created"  type = list(object({  path = string  capabilities = set(string)  }))  } *[Incorrect Code (Wrong Logic), Inconsistent (Code)]*  ```    With this change, `var.policy` will be treated as a list of objects instead of a map of lists.    Additionally, you'll need to update the dynamic block to use the `index` function to access the object attributes. Here's the updated code: *[Incorrect (Concept), Inconsistent (Concept)]*    ```terraform  data "vault\_policy\_document" "this" {  dynamic "rule" {  for\_each = {  for idx, p in var.policy : format("%s-%s", p.path, join(",", p.capabilities)) => idx  }  content {  path = var.policy[index(var.policy, rule.value)].path  capabilities = var.policy[index(var.policy, rule.value)].capabilities  }  }  } *[Incorrect Code (Wrong Logic), Inconsistent (Code)]*  ```    In the updated code, the `index(var.policy, rule.value)` expression is used to retrieve the object at the given index in the `var.policy` list. This allows you to access the `path` and `capabilities` attributes of each object correctly. *[Not Concise (Redundant)]*    After making these changes, the dynamic block should work as expected with your list of objects in `var.policy`. *[Not Concise (Redundant)]*  I hope this helps! Let me know if you have any further questions. *[Not Concise (Excess)]* | Fully Comprehensive  Usefulness (2)  -------------------  Incorrect  Inconsistent  Not Concise |
| 75299506 | cannot import name 'save\_virtual\_workbook' from 'openpyxl.writer.excel'  <https://stackoverflow.com/questions/75299506/cannot-import-name-save-virtual-workbook-from-openpyxl-writer-excel> | The `save\_virtual\_workbook` function is not available in the `openpyxl` library. It seems you might be referring to a function from a different library or using an outdated code snippet.    In the `openpyxl` library, you can use the `save` function to save a workbook to a file. Here's an example:    ```python  from openpyxl import Workbook    # Create a workbook  workbook = Workbook()    # Do some operations on the workbook    # Save the workbook to a file  workbook.save("output.xlsx")  ```    If you are specifically looking to save the workbook to a virtual file-like object, you can use the `BytesIO` class from the `io` module to create a virtual file-like object and save the workbook to it. Here's an example: *[Inconsistent (Concept), Inconsistent (Number of Solutions)]*    ```python  from openpyxl import Workbook  from io import BytesIO    # Create a workbook  workbook = Workbook()    # Do some operations on the workbook    # Save the workbook to a virtual file-like object  virtual\_file = BytesIO()  workbook.save(virtual\_file) *[Inconsistent (Code)]*  ```    In the above example, `virtual\_file` is a `BytesIO` object that can be used as a file-like object to read or write binary data. *[Not Concise (Redundant)]*    Make sure you have the latest version of the `openpyxl` library installed by running `pip install --upgrade openpyxl`. If you still encounter issues, please provide more details or share the code snippet where you're experiencing the problem. *[Not Concise (Irrelevant)]* | Fully Comprehensive  Usefulness (5)  -------------------  Correct  Inconsistent  Partially Concise |
| 75303196 | Setting an array element with a sequence.The requested array has an inhomogeneous shape after1dimensions.The detected shapewas(18,)+inhomogeneouspart  <https://stackoverflow.com/questions/75303196/setting-an-array-element-with-a-sequence-the-requested-array-has-an-inhomogeneou> | The error message you're encountering typically occurs when you try to assign a sequence (such as a list) to an array element, but the shapes of the elements are not compatible.    In your code, it appears that the `Age\_list` and `Lactate\_list` are lists, not NumPy arrays. *[Incorrect (Factual), Inconsistent (Factual)]*  When you try to pass these lists to `np.polyfit` or use them in plotting functions, you might encounter the error.    To resolve this issue, you can convert your lists to NumPy arrays using `np.array()`. Here's an updated version of your code: *[Incorrect (Factual), Inconsistent (Factual)]*    ```python  import numpy as np  import matplotlib.pyplot as plt    def plot():  Age\_list = []  Lactate\_list = []  for l in range(1, 19):  Age = test[test.ID == l]['age']  Lactate = (test[test.ID == l]['VO2'].nlargest(n=5).mean()) \* (80 / 100)  Lactate\_list.append(Lactate)  Age\_list.append(Age)    Age\_array = np.array(Age\_list)  Lactate\_array = np.array(Lactate\_list)    plt.scatter(Age\_array, Lactate\_array, color='purple')  a, b = np.polyfit(Age\_array, Lactate\_array, 1)  plt.plot(Age\_array, a \* Age\_array + b)  plt.xlabel('Age')  plt.ylabel('Lactate threshold')  plt.title('Correlation between Age and Lactate threshold')  plt.show() *[Incorrect Code (Wrong Logic), Inconsistent (Code)]*  ```    In this updated code, I've converted `Age\_list` and `Lactate\_list` to NumPy arrays using `np.array()`, and then used these arrays in the scatter plot and `np.polyfit` function. This should help resolve the error you're encountering. *[Incorrect (Factual), Inconsistent (Factual)]*    Make sure you have the necessary dependencies (`numpy` and `matplotlib`) installed in your Python environment. If the error persists, please provide more details or a sample of the input data so that I can assist you further. *[Not Concise (Irrelevant)]* |  |
| 75309076 | TypeError: \_\_init\_\_() takes 2 positional arguments but 3 were given - feature extraction using recursive feature elimination  <https://stackoverflow.com/questions/75309076/typeerror-init-takes-2-positional-arguments-but-3-were-given-feature-ex> | The error you're encountering is due to the incorrect usage of the `RFE` class in scikit-learn's feature selection module. The `RFE` class does not take the number of features as a positional argument; instead, it expects it to be passed as a named argument `n\_features\_to\_select`.    To fix the error, modify the line where you initialize `rfe` as follows:    ```python  rfe = RFE(estimator=model, n\_features\_to\_select=10)  ```    Here, the `estimator` argument should be set to the model you want to use for feature selection (`model` in your case), and `n\_features\_to\_select` specifies the number of features to retain.    After making this change, the error should be resolved, and the feature extraction using Recursive Feature Elimination should work as expected.    Here's the updated code:    ```python  from sklearn.feature\_selection import RFE  from sklearn.linear\_model import LogisticRegression  # ...  model = LogisticRegression(solver='lbfgs')  rfe = RFE(estimator=model, n\_features\_to\_select=10)  fit = rfe.fit(X, Y)  print("Num Features: %s" % (fit.n\_features\_))  print("Selected Features: %s" % (fit.support\_))  print("Feature Ranking: %s" % (fit.ranking\_))  ```    Make sure you have scikit-learn installed (`pip install scikit-learn`) and import the necessary modules (`from sklearn.feature\_selection import RFE, from sklearn.linear\_model import LogisticRegression`) at the beginning of your script.    Let me know if you have any further questions! *[Not Concise (Irrelevant)]* | Fully Comprehensive  Usefulness (5)  -------------------  Correct  Consistent  Not Concise |
| 75322813 | OpenAI GPT-3 API error: "That model does not exist"  <https://stackoverflow.com/questions/75322813/openai-gpt-3-api-error-that-model-does-not-exist> | The error message "That model does not exist" typically occurs when you are trying to access a model that is either invalid or not available in the OpenAI API. In your case, it seems that the `chat-gpt` engine may not be recognized as a valid engine.    To resolve this issue, you can check the available engines and ensure that you are using the correct engine name. As of my knowledge cutoff in September 2021, the available engines are `davinci`, `curie`, `babbage`, `ada`, and `text-davinci-003`.    You can update your code to use one of the available engines. For example, you can replace `chat-gpt` with `davinci` as follows:    ```javascript  const chatGptUrl = "https://api.openai.com/v1/engines/davinci/completions";  // ...  ```    Make sure you have the correct engine name and try running your code again.    If the issue persists, please check the OpenAI API documentation or contact OpenAI support for further assistance. *[Not Concise (Irrelevant)]* | Fully Comprehensive  Usefulness (5)  -------------------  Correct  Consistent  Not Concise |
| 75324072 | Pandas JSON Orient Autodetection  <https://stackoverflow.com/questions/75324072/pandas-json-orient-autodetection> | Yes, `pd.read\_json` in Pandas performs some level of autodetection to determine the orientation of the JSON data.    By default, when you pass a JSON array (such as `data\_records` in your example) to `pd.read\_json`, it assumes the orientation is "columns" and interprets each object in the array as a column. This is why you get the desired output with the data in record-based format.    When you pass a JSON object (such as `data\_index` in your example), `pd.read\_json` automatically detects that the orientation is "index" and interprets the keys of the object as the index labels. This is why you get the expected output with the data in column-based format.    You can also explicitly specify the orientation using the `orient` parameter of `pd.read\_json`. By setting `orient="index"`, you inform Pandas that the JSON data should be treated as index-based.    In summary, `pd.read\_json` does perform autodetection to determine the orientation of the JSON data, but you can also explicitly specify the orientation using the `orient` parameter if needed. *[Not Concise (Redundant)]* | Fully Comprehensive  Usefulness (5)  -------------------  Correct  Consistent  Not Concise |
| 75329242 | Installing hermes-engine (0.71.2) got stuck react native  <https://stackoverflow.com/questions/75329242/installing-hermes-engine-0-71-2-got-stuck-react-native> | It seems that the installation process is stuck at the "hermes-engine" package. This can happen due to various reasons, such as network issues or conflicts with other dependencies. *[Incorrect (Factual), Inconsistent (Factual)]*    Here are a few steps you can try to resolve the issue: *[Inconsistent (Number of Solutions)]*    1. Cancel the installation process: Press Ctrl+C in the terminal to stop the current installation. This will allow you to proceed with troubleshooting.    2. Clear npm cache: Run the following command to clear the npm cache:  ```css  npm cache clean --force  ```    3. Delete node\_modules: Remove the "node\_modules" directory by running the following command in the project's root directory:  ```bash  rm -rf node\_modules  ```    4. Update npm and Node.js: Ensure that you have the latest version of npm and Node.js installed. You can update npm by running:  ```  npm install -g npm  ```  Make sure you have a compatible version of Node.js installed as per your React Native version.    5. Check your network connection: Ensure that you have a stable internet connection. Sometimes, a slow or intermittent connection can cause issues during package installation.    6. Disable IPv6: If you're using IPv6, try disabling it temporarily and then attempt the installation again. Some users have reported that disabling IPv6 resolved their installation issues.    7. Try a different package manager: Instead of using npm, you can try using Yarn to install the dependencies. First, remove the "node\_modules" directory, then run:  ```  yarn install  ```    8. Install hermes-engine separately: If the above steps don't work, you can try installing the "hermes-engine" package separately using the following command:  ```  npm install hermes-engine *[Incorrect (Concept), Inconsistent (Concept)]*  ```    After trying these steps, you can retry the `pod install` command to complete the installation of the iOS dependencies.    If the issue persists, it may be helpful to provide any error messages or logs that you encounter during the installation process. *[Not Concise (Excess)]* | Not Comprehensive  Usefulness (1)  -------------------  Incorrect  Inconsistent  Partially Concise |
| 75331454 | Gradle 7.4 build with resolve error: The server may not support the client's requested TLS protocol versions  <https://stackoverflow.com/questions/75331454/gradle-7-4-build-with-resolve-error-the-server-may-not-support-the-clients-req> | The error message suggests that the server hosting the Joda-Time library may not support the TLS protocol versions (TLSv1.1, TLSv1.2, TLSv1.3) requested by the client (Gradle). *[Incorrect (Factual), Inconsistent (Factual)]*    To resolve this issue, you can try the following steps: *[Inconsistent (Number of Solutions)]*    1. Update Gradle properties: Remove the explicit TLS protocol versions from your Gradle properties file. Change the following lines:  ```  systemProp.https.protocols=TLSv1.1,TLSv1.2,TLSv1.3  systemProp.jdk.tls.client.protocols=TLSv1.1,TLSv1.2,TLSv1.3  ```  to:  ```  systemProp.https.protocols=TLSv1.2,TLSv1.3  systemProp.jdk.tls.client.protocols=TLSv1.2,TLSv1.3  ```    2. Set Java system properties: Instead of setting the TLS protocols through Gradle properties, you can set them as Java system properties. Modify your build command to include the following system properties:  ```  ./gradlew -Dhttps.protocols=TLSv1.2,TLSv1.3 -Djdk.tls.client.protocols=TLSv1.2,TLSv1.3 <build-command>  ```    For example, if you're running the `assemble` task, the command would be:  ```  ./gradlew -Dhttps.protocols=TLSv1.2,TLSv1.3 -Djdk.tls.client.protocols=TLSv1.2,TLSv1.3 assemble  ```    This ensures that Gradle and Java use the specified TLS protocol versions when making network requests.    3. Retry the build: After making the necessary changes, try running the build command again and see if the issue persists. *[Incorrect (Concept), Inconsistent (Concept)]*    If the problem still occurs, it's possible that the server hosting the Joda-Time library has compatibility issues with the TLS protocol versions you're using. In such cases, you may need to contact the maintainers of the library or try using a different version that is compatible with your Gradle and JDK setup. | Fully Comprehensive  Usefulness (1)  -------------------  Incorrect  Inconsistent  Not Concise |