Android Studio

Code Validation and Quick Fixes for Endpoints Backends

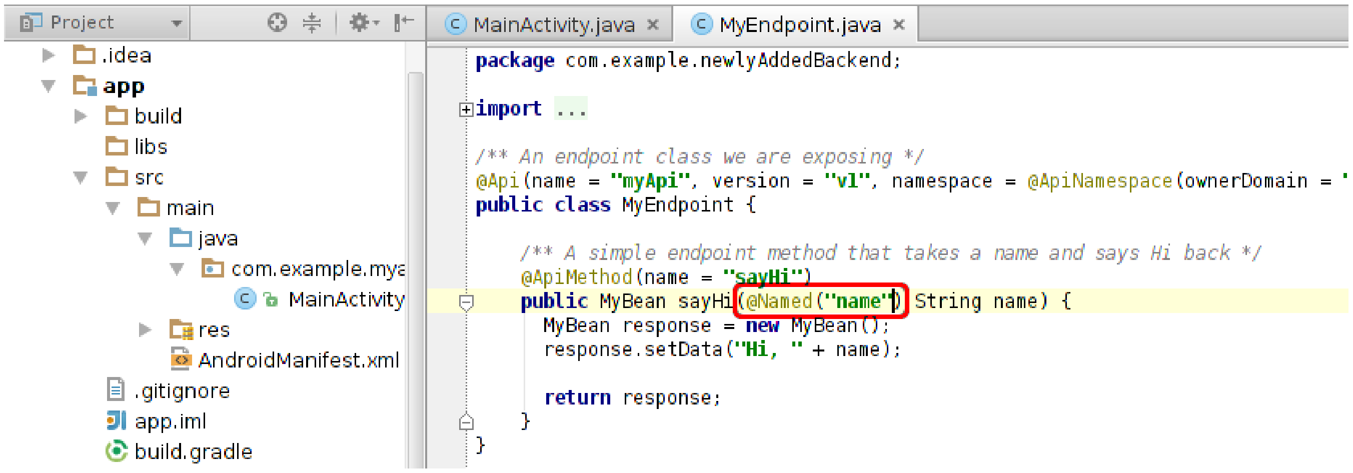
Server-side Endpoints API definitions must conform to a number of syntactic rules. Android Studio knows these and validates your code as you type to help you avoid making mistakes. Android Studio provides Endpoints-specific inspections and quick-fixes.

As-you-type code validation

For example, the default backend type *App Engine Java Endpoints Module* contains the following minimal annotated Endpoints API located in your project at `<backend-name>/src/main/java/<package-name>/MyEndpoint.java`:

```
import javax.inject.Named;

@Api(name = "myApi", version = "v1",
     namespace = @ApiNamespace(ownerDomain = "<package-name>",
                               ownerName = "<package-name>",
                               packagePath=""))
public class MyEndpoint {
    @ApiMethod(name = "sayHi")
    public MyBean sayHi(@Named("name") String name) {
      MyBean response = new MyBean();
      response.setData("Hi, " + name);

      return response;
    }
}
```

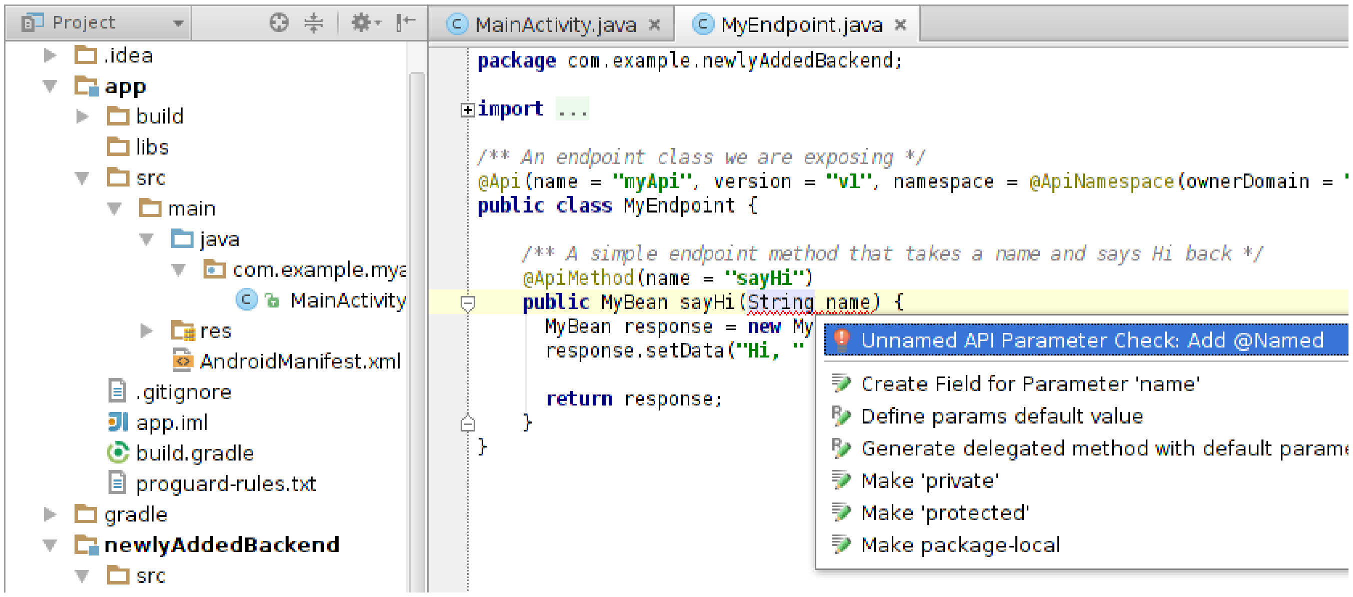
In the code, `@Named` is required for all non-entity type parameters passed to server-side methods. If you forget to add this annotation when modifying `sayHi` in this code, Android Studio will underline the problematic statement as you type, as shown below:



```
package com.example.newlyAddedBackend;

import ...

/** An endpoint class we are exposing */
@Api(name = "myApi", version = "v1", namespace = @ApiNamespace(ownerDomain = '
public class MyEndpoint {

    /** A simple endpoint method that takes a name and says Hi back */
    @ApiMethod(name = "sayHi")
    public MyBean sayHi(@Named("name") String name) {
        MyBean response = new MyBean();
        response.setData("Hi, " + name);

        return response;
    }
}
```

Quick fixes

To help you avoid some of the more common Cloud Endpoints development mistakes, Android Studio provides quick fixes. To see these quick fix suggestions, press **Alt** + **Enter** if you're running on Linux/Windows, or ⌥ + **Enter** if you're running on a Mac. For example, here is Android Studio displaying a quick fix for the missing `@Named` annotation shown in the above code sample:



```
package com.example.newlyAddedBackend;

import ...

/** An endpoint class we are exposing */
@Api(name = "myApi", version = "v1", namespace = @ApiNamespace(ownerDomain = '
public class MyEndpoint {

    /** A simple endpoint method that takes a name and says Hi back */
    @ApiMethod(name = "sayHi")
    public MyBean sayHi(String name) {
        MyBean response = new My
        response.setData("Hi, "

        return response;
    }
}
```

As expected, choosing the first quick fix (`Add @Named`) adds `@Named` to the method parameter.

Last updated December 4, 2019.