

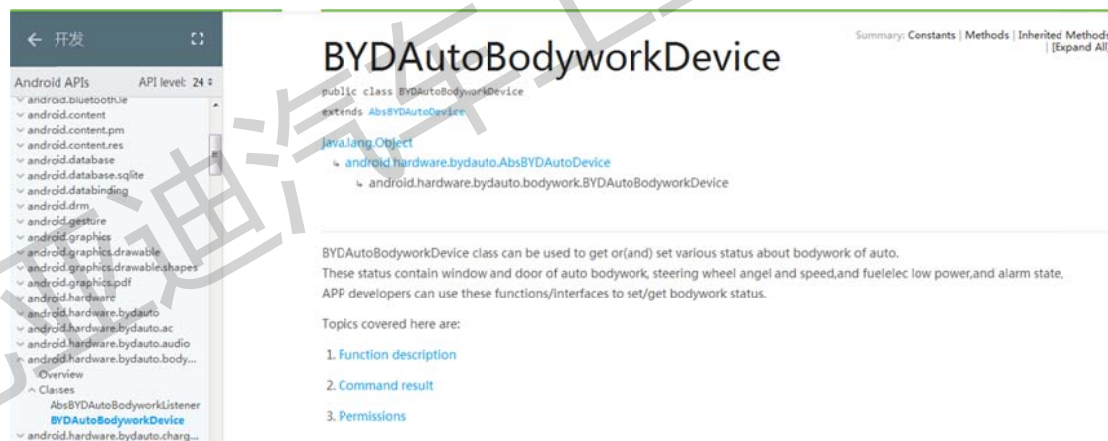
使用 BYD-AUTO-API 的 android 示例

一.简介

关于 APK 如何使用 BYD-AUTO-API

二.Code

当我们装好 SDK 过后，在其目录下会有一个 docs 文件夹，打开该文件夹，可以找到一个 index.html 文件，点击打开查看描述。DEVELOP -> Reference

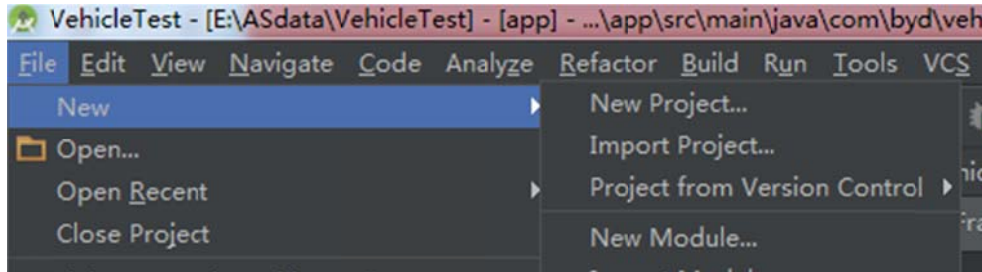


Function description

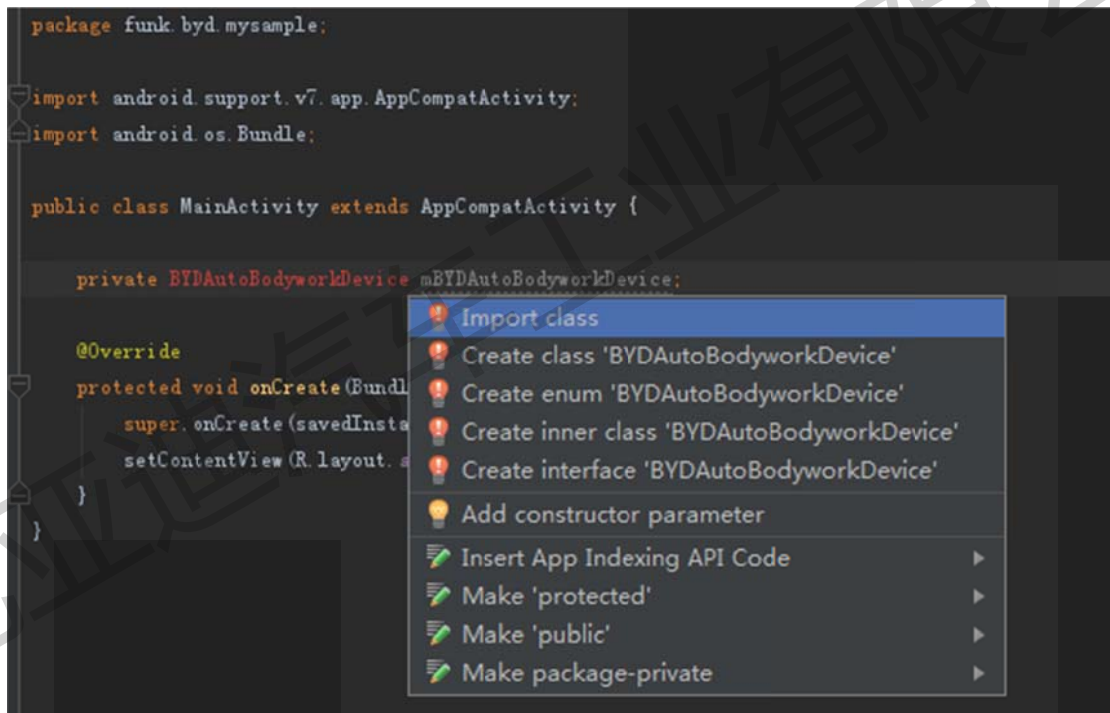
- The class first needs to be instantiated using the `getInstance(Context)` method;
- Register and unregister bodywork listener:
 - If we want to listen the underlying window/door state's changes, we should use `registerListener(AbsBYDAutoBodyworkListener)` method to register bodywork listener first.
 - Next use the callback method like `onWindowStateChange(int, int)` to receive the change of window state of bodywork.
 - We should remember to unregister the listener through `unregisterListener(AbsBYDAutoBodyworkListener)` method when it is no longer used.
- Then we can set status of bodywork such as window state through like `setXxx()` method and get status such as door state through like `getXxx()` method.

1. 新建工程

打开 Android studio 新建工程



在自己的 MainActivity（或数据层）定义属性，比如现在我们要获取车身状态类相关信息，需要一个 BYDAutoBodyworkDevice 的实例；



引入 BYDAutoBodyworkDevice 类 需要安装了 BYD-AUTO-API 的 SDK ,才能导入此包；

```
import android.hardware.bydauto.bodywork.BYDAutoBodyworkDevice;
```

在导入成功后，需要获取该实例，由文档 index.html 里的描述得知，使用

getInstance(Context)获取，需要一个上下文，我们传入此应用的 Context 成功获取到

BYDAutoBodyworkDevice 实例；

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    mBYDAutoBodyworkDevice = BYDAutoBodyworkDevice.getInstance(getApplicationContext());
}
```

2. 获取函数(get)

在文档 index.html 里，描述了所有函数/接口，例如接下来我们获取 VIN 号；

java.lang.String

getAutoVIN()
Use this method to get the vin of auto.

Permissions

- When getxxx() method is used, you need to declare permission android.permission.BYDAUTO_BODYWORK_GET <uses-permission>.

可以找到此函数，执行可返回 String 类型的 VIN 号，但说明文档提示我们需要声明一个权限，所以我们在清单文件 AndroidManifest.xml 里添加；

```
<uses-permission
android:name="android.permission.BYDAUTO_BODYWORK_GET" />
```

(还有个新权限可能需要添加)

```
<uses-permission
android:name="android.permission.BYDAUTO_BODYWORK_COMMON" />
```

然后定义自己的 UI 显示，输出返回的 String，这里用一个 TextView 表示；

```
private TextView mTextView;
```

```
@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);

    mBydAutoBodyworkDevice =
BYDAutoBodyworkDevice.getInstance(getApplicationContext());

    mTextView = (TextView) findViewById(R.id.text);
}

@Override

protected void onResume() {

    super.onResume();

    String autoVIN = mBydAutoBodyworkDevice.getAutoVIN();

    mTextView.setText(autoVIN);
}
```

3.设置函数(set)

Set 方法与 get 方法类似，把需要设置的参数传入即可

4.监听器(Listener)

监听的作用是在值发生改变的时候 通知 UI 更新或者做相应的处理 如监听电源档位变化 ,

首先查看文档 ;

Summary: Constants | Ctors | Methods | Inherited Methods | [Expand All]

AbsBYDAutoBodyworkListener

public abstract class AbsBYDAutoBodyworkListener
extends [Object](#) implements [IBYDAutoListener](#)

[java.lang.Object](#)

- [android.hardware.bydauto.bodywork.AbsBYDAutoBodyworkListener](#)

onPowerLevelChanged

void onPowerLevelChanged (int level)

Receive bodywork power level changed message.

Parameters	
level	int: This param is the current auto power level. the level is: OFF level: BODYWORK_POWER_LEVEL_OFF ACC level: BODYWORK_POWER_LEVEL_ACC ON level: BODYWORK_POWER_LEVEL_ON OK level: BODYWORK_POWER_LEVEL_OK

See also:

[getPowerLevel\(\)](#)

需要获得 AbsBYDAutoBodyworkListener 实例 ;

Public constructors

[AbsBYDAutoBodyworkListener\(\)](#)

有提供的构造方法 , 所以 ;

```
private BYDAutoBodyworkDevice mBydAutoBodyworkDevice;

private AbsBYDAutoBodyworkListener mAbsBYDAutoBodyworkListener;

private TextView mTextView;

@Override
```

```
protected void onCreate(Bundle savedInstanceState) {  
  
    super.onCreate(savedInstanceState);  
  
    setContentView(R.layout.activity_main);  
  
    mBydAutoBodyworkDevice =  
BYDAutoBodyworkDevice.getInstance(getApplicationContext());  
  
    mTextView = (TextView) findViewById(R.id.text);  
  
    mAbsBYDAutoBodyworkListener = new AbsBYDAutoBodyworkListener() {  
  
        @Override  
  
        public void onPowerLevelChanged(int level) {  
  
            super.onPowerLevelChanged(level);  
  
            mTextView.setText(level);  
  
        }  
  
    };  
}
```

重写 onPowerLevelChanged()，在此代码块实现业务逻辑即可，如图重新设置 TextView 显示，得到监听器后还需要注册监听器；

```
@Override  
  
protected void onResume() {  
  
    super.onResume();  
  
}
```

```
String autoVIN = mBydAutoBodyworkDevice.getAutoVIN();

mTextView.setText(autoVIN);

mBydAutoBodyworkDevice.registerListener(mAbsBYDAutoBodyworkListener);
}

@Override
protected void onPause() {

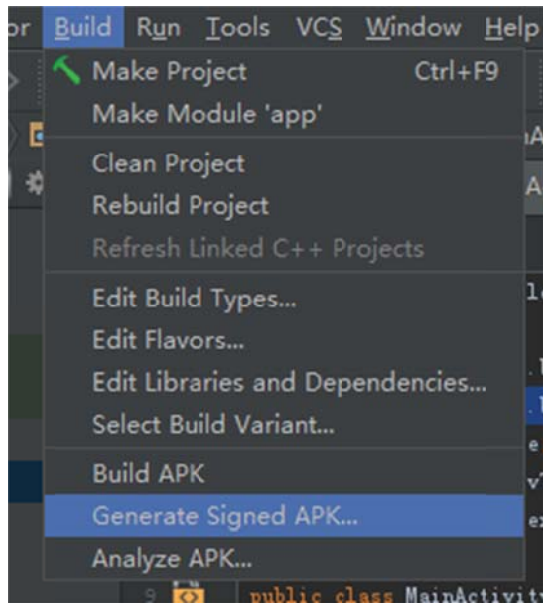
    super.onPause();

    mBydAutoBodyworkDevice.unregisterListener(mAbsBYDAutoBodyworkListener);
}
```

同时在 Activity 暂停时解绑监听器，这样在 Activity 的生命周期内能够监听到电源档位的
变化。

三.生成 APK

生成 APK 需要签名对应，才能安装



比亚迪汽车工业有限公司