

# 访问互联网

---

## 1 常用网络协议

### 1.1 超文本传输协议(HyperText Transfer Protocol, HTTP)

HTTP协议是互联网技术中运用最为广泛的应用层协议。在使用浏览器访问网站时，就是通过HTTP协议的URL进行访问的。

### 1.2 传输控制协议(Transmission Control Protocol, TCP)

TCP协议是可靠的传输层协议。该协议可以在联网设备之间建立稳定的连接，通常用于对数据准确性要求较高，且传输量不大的场合。诸如网络游戏、社交软件中常能见到TCP协议的身影。

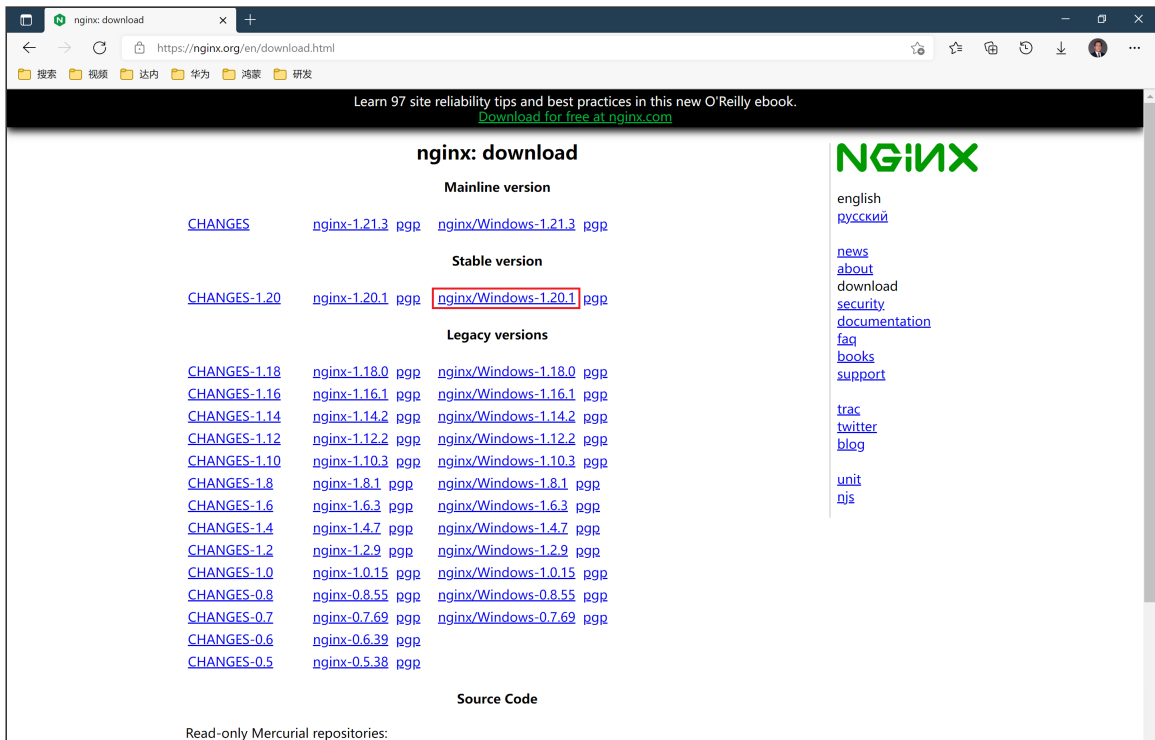
### 1.3 用户数据报文协议(User Datagram Protocol, UDP)

UDP协议是无连接、不可靠的传输层协议。基于UDP协议的网络通信，不需要建立连接，速度更快，占用资源更少，程序结构也更简单，通常用于对数据准确性要求不高，且传输量较大的场合，如网络直播、视频点播等。

## 2 搭建Web服务器

### 2.1 下载安装Nginx

<https://nginx.org/en/download.html>

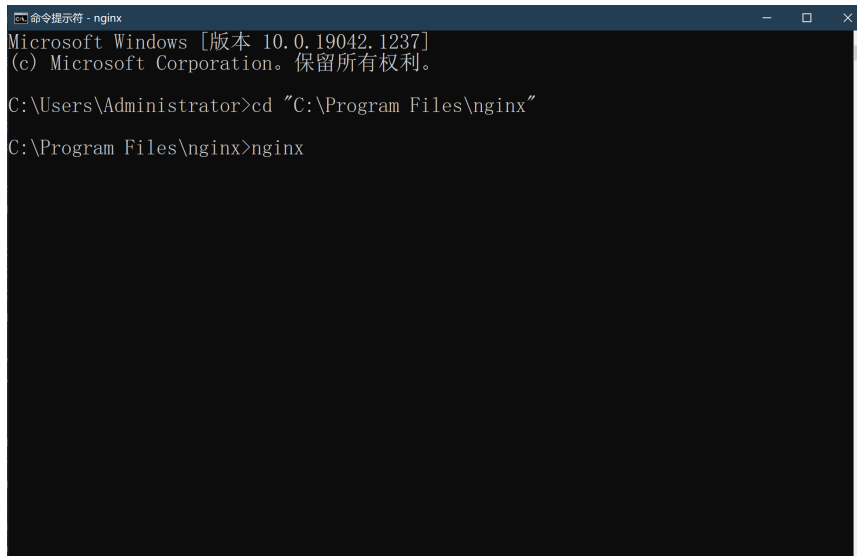


Stable version: nginx/Windows-1.20.1

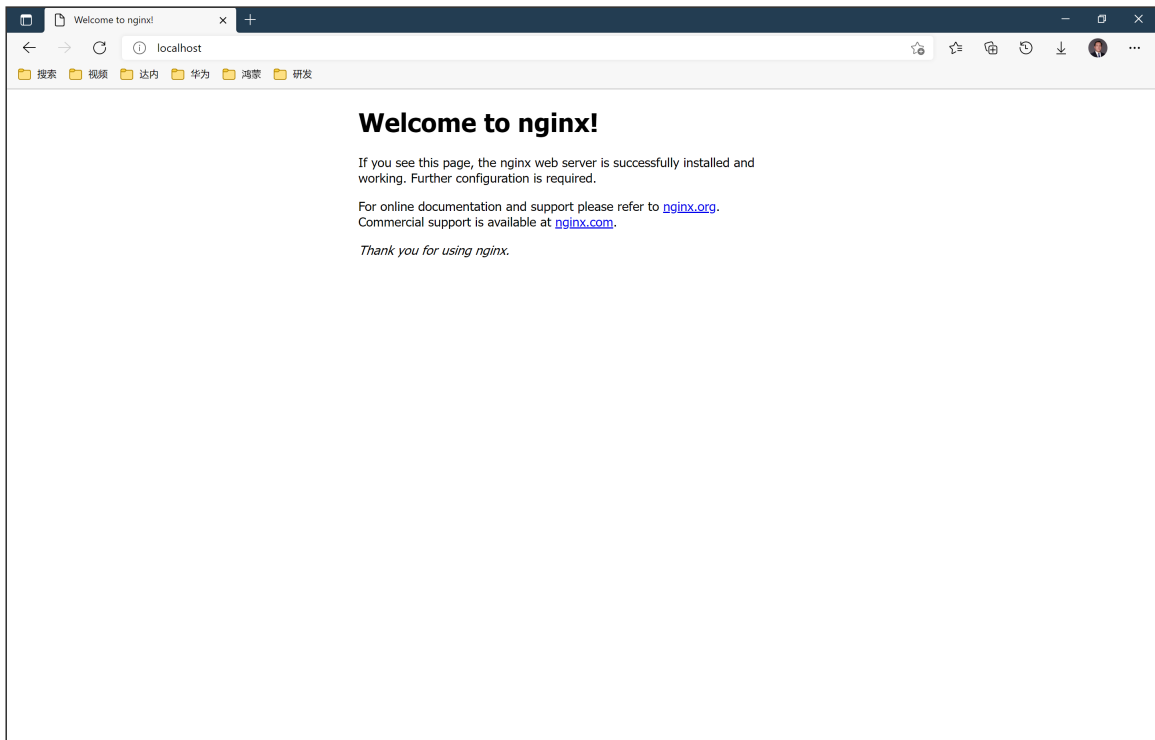
解压到C:\Program Files\nginx目录下。

## 2.2 启动运行Nginx

```
C:\Program Files\nginx>nginx
```



浏览器: <http://localhost>



## 2.3 通过IP地址访问Nginx

```
C:\Users\Administrator>ipconfig
```

```
命令提示符
C:\Users\Administrator>ipconfig

Windows IP 配置

无线局域网适配器 本地连接* 1:

    媒体状态 . . . . . : 媒体已断开连接
    连接特定的 DNS 后缀 . . . . . :

无线局域网适配器 本地连接* 2:

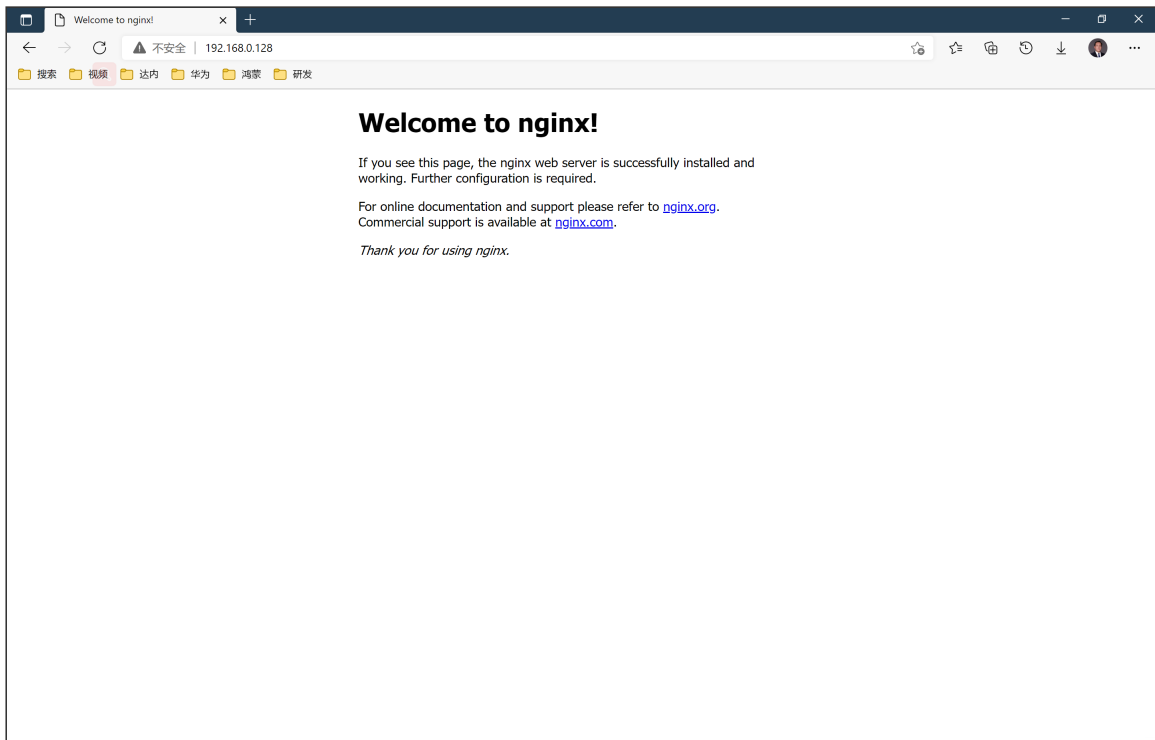
    媒体状态 . . . . . : 媒体已断开连接
    连接特定的 DNS 后缀 . . . . . :

无线局域网适配器 WLAN:

    连接特定的 DNS 后缀 . . . . . :
    本地链接 IPv6 地址. . . . . : fe80::158b:9610:5086:4f4d%14
    IPv4 地址 . . . . . : 192.168.0.128
    子网掩码 . . . . . : 255.255.255.0
    默认网关. . . . . : 192.168.0.1

以太网适配器 蓝牙网络连接:
```

浏览器: <http://192.168.0.128>

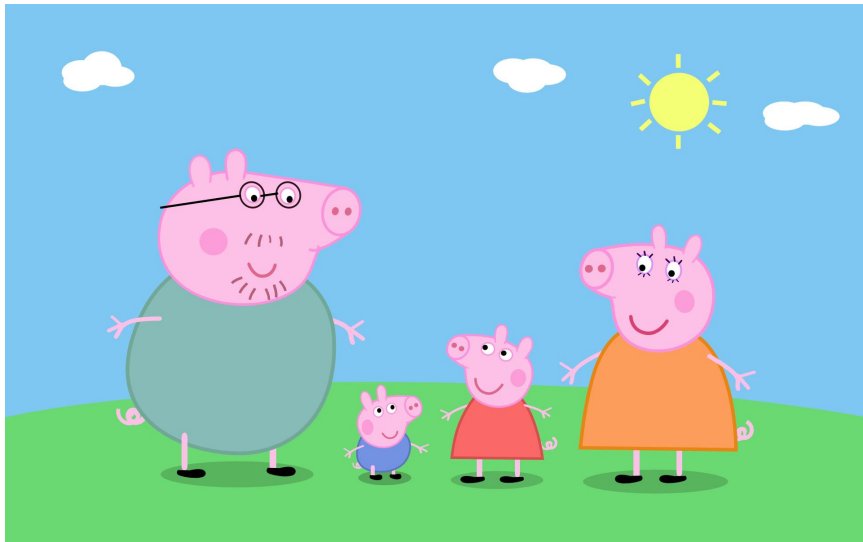


## 2.4 添加网站内容

C:\Program Files\nginx\html\t\_student.json

```
[
  {
    "id": 1,
    "name": "张飞",
    "age": 22,
    "sex": 1,
    "subject": "C++"
  },
  {
    "id": 2,
    "name": "貂蝉",
    "age": 18,
    "sex": 0,
    "subject": "Java"
  },
  {
    "id": 3,
    "name": "曹操",
    "age": 30,
    "sex": 1,
    "subject": "Java"
  }
]
```

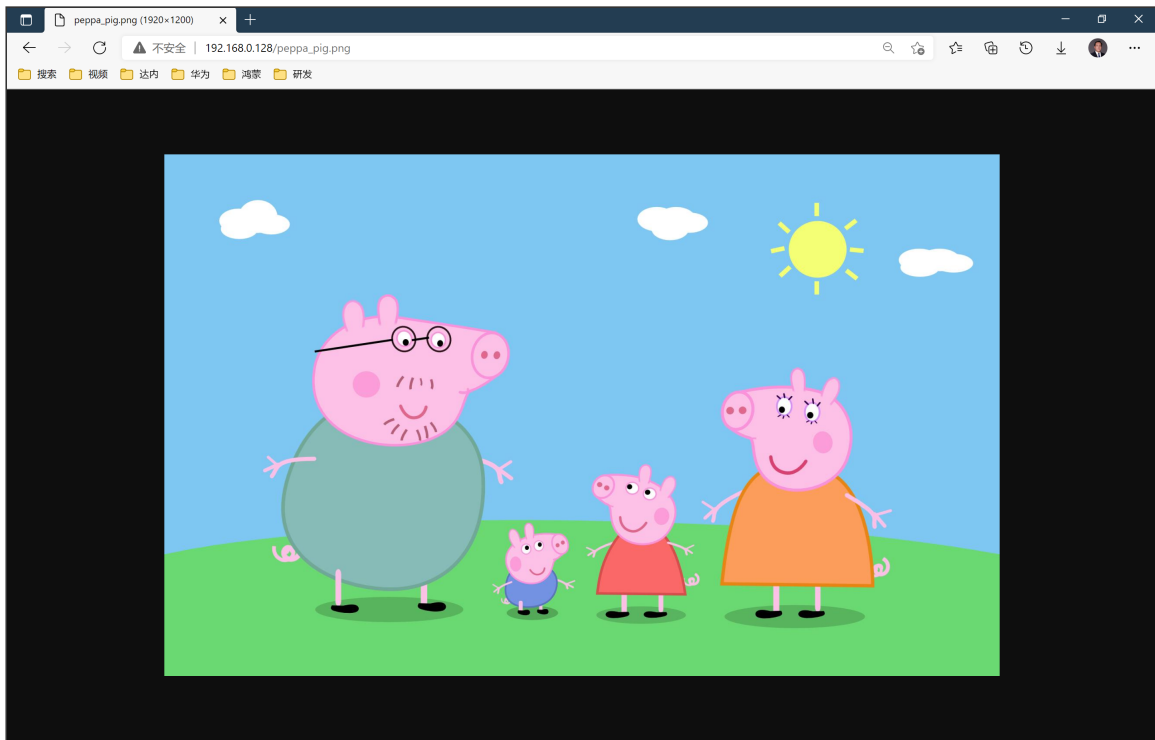
C:\Program Files\nginx\html\peppa\_pig.png



浏览器: [http://192.168.0.128/t\\_student.json](http://192.168.0.128/t_student.json)

```
[
  {
    "id": 1,
    "name": "张飞",
    "age": 22,
    "sex": 1,
    "subject": "C++"
  },
  {
    "id": 2,
    "name": "貂蝉",
    "age": 18,
    "sex": 0,
    "subject": "Java"
  },
  {
    "id": 3,
    "name": "曹操",
    "age": 30,
    "sex": 1,
    "subject": "Java"
  }
]
```

浏览器: [http://192.168.0.128/peppa\\_pig.png](http://192.168.0.128/peppa_pig.png)



## 3 通过Http协议访问Web服务器

### 3.1 添加权限并将Http设置为Web访问的缺省模式

config.json

```
{
  ...
  "deviceConfig": {
    "default": {
      "network": {
        "cleartextTraffic": true
      }
    }
  },
  "module": {
    ...
    "reqPermissions": [
      {
        "name": "ohos.permission.INTERNET"
      }
    ]
  }
}
```

### 3.2 创建工作线程

```
getGlobalTaskDispatcher(TaskPriority.DEFAULT)
    .asyncDispatch(() -> getText());

getGlobalTaskDispatcher(TaskPriority.DEFAULT)
    .asyncDispatch(() -> getImage());
```

### 3.3 从Web服务器获取文本

```
try {
    URL url = new URL("http://192.168.0.128/t_student.json");
    URLConnection connection = url.openConnection();

    InputStream is = connection.getInputStream();
    InputStreamReader isr = new InputStreamReader(is);
    BufferedReader br = new BufferedReader(isr);

    String json = "", line;
    while ((line = br.readLine()) != null)
        json += line;

    ZSONArray students = ZSONArray.stringToZSONArray(json);
    for (int i = 0; i < students.size(); ++i) {
        ZSONObject student = students.getZSONObject(i);

        int id = student.getInteger("id");
        String name = student.getString("name");
        int age = student.getInteger("age");
        int sex = student.getInteger("sex");
        String subject = student.getString("subject");

        HiLog.info(label, "%{public}d, %{public}s, " +
            "%{public}d, %{public}d, %{public}s",
            id, name, age, sex, subject);
    }
}
catch (Exception exception) {
    HiLog.info(label, exception.getLocalizedMessage());
}
```

### 3.4 从Web服务器获取图像

```

try {
    URL url = new URL("http://192.168.0.128/peppa_pig.png");
    URLConnection connection = url.openConnection();

    InputStream is = connection.getInputStream();
    ImageSource source = ImageSource.create(is,
        new SourceOptions());
    DecodingOptions options = new DecodingOptions();
    PixelMap map = source.createPixelmap(options);

    int width = map.getImageInfo().size.width;
    int height = map.getImageInfo().size.height;
    HiLog.info(label, "图像宽度%{public}d像素, " +
        "图像高度%{public}d像素", width, height);
}
catch (Exception exception) {
    HiLog.info(label, exception.getLocalizedMessage());
}

```

例程: Http

...\Http\entry\src\main\config.json

```

{
    ...
    "deviceConfig": {
        "default": {
            "network": {
                "cleartextTraffic": true
            }
        }
    },
    "module": {
        ...
        "reqPermissions": [
            {
                "name": "ohos.permission.INTERNET"
            }
        ]
    }
}

```

...\Http\entry\src\main\resources\base\graphic\background\_button.xml

```

<?xml version="1.0" encoding="utf-8"?>
<shape
    xmlns:ohos="http://schemas.huawei.com/res/ohos"
    ohos:shape="oval">
    <solid ohos:color="#00a2e8"/>
</shape>

```



...\Http\entry\src\main\resources\base\layout\ability\_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<DirectionalLayout
  xmlns:ohos="http://schemas.huawei.com/res/ohos"
  ohos:height="match_parent"
  ohos:width="match_parent"
  ohos:alignment="center"
  ohos:orientation="vertical"
  ohos:background_element="#004080">

  <Button
    ohos:id="$+id:btnText"
    ohos:height="96vp"
    ohos:width="96vp"
    ohos:background_element="$graphic:background_button"
    ohos:text="文本"
    ohos:text_size="24fp"
    ohos:text_color="#ffffff"
  />

  <Button
    ohos:id="$+id:btnImage"
    ohos:height="96vp"
    ohos:width="96vp"
    ohos:top_margin="24vp"
    ohos:background_element="$graphic:background_button"
    ohos:text="图像"
    ohos:text_size="24fp"
    ohos:text_color="#ffffff"
  />

</DirectionalLayout>
```

...\Http\entry\src\main\java\com\minwei\http\slice>MainAbilitySlice.java

```

public class MainAbilitySlice extends AbilitySlice {
    private static final HiLogLabel label = new HiLogLabel(
        HiLog.LOG_APP, 0x00101,
        MainAbilitySlice.class.getCanonicalName());

    @Override
    public void onStart(Intent intent) {
        ...
        findViewById(ResourceTable.Id_btnText)
            .setClickListener(component -> onText());
        findViewById(ResourceTable.Id_btnImage)
            .setClickListener(component -> onImage());
    }
    ...
    private void onText() {
        getGlobalTaskDispatcher(TaskPriority.DEFAULT)
            .asyncDispatch(() -> getText());
    }

    private void onImage() {
        getGlobalTaskDispatcher(TaskPriority.DEFAULT)
            .asyncDispatch(() -> getImage());
    }

    private void getText() {
        try {
            URL url = new URL("http://192.168.0.128/t_student.json");
            URLConnection connection = url.openConnection();

            InputStream is = connection.getInputStream();
            InputStreamReader isr = new InputStreamReader(is);
            BufferedReader br = new BufferedReader(isr);

            String json = "", line;
            while ((line = br.readLine()) != null)
                json += line;

            ZSONArray students = ZSONArray.stringToZSONArray(json);
            for (int i = 0; i < students.size(); ++i) {
                ZSONObject student = students.getZSONObject(i);

                int id = student.getInteger("id");
                String name = student.getString("name");
                int age = student.getInteger("age");
                int sex = student.getInteger("sex");
                String subject = student.getString("subject");

                HiLog.info(label, "%{public}d, %{public}s, " +
                    "%{public}d, %{public}d, %{public}s",
                    id, name, age, sex, subject);
            }
        }
        catch (Exception exception) {
            HiLog.info(label, exception.getLocalizedMessage());
        }
    }
}

```

```
    }  
}  
  
private void getImage() {  
    try {  
        URL url = new URL("http://192.168.0.128/peppa_pig.png");  
        URLConnection connection = url.openConnection();  
  
        InputStream is = connection.getInputStream();  
        ImageSource source = ImageSource.create(is,  
            new SourceOptions());  
        DecodingOptions options = new DecodingOptions();  
        PixelMap map = source.createPixelmap(options);  
  
        int width = map.getImageInfo().size.width;  
        int height = map.getImageInfo().size.height;  
        HiLog.info(label, "图像宽度%{public}d像素, " +  
            "图像高度%{public}d像素", width, height);  
    }  
    catch (Exception exception) {  
        HiLog.info(label, exception.getLocalizedMessage());  
    }  
}  
}
```

运行效果如下图所示:



```
dependencies {
    ...
    implementation 'com.squareup.okhttp3:okhttp:4.9.0'
}
```

## 4.2 添加权限并将Http设置为Web访问的缺省模式

config.json

```
{
    ...
    "deviceConfig": {
        "default": {
            "network": {
                "cleartextTraffic": true
            }
        }
    },
    "module": {
        ...
        "reqPermissions": [
            {
                "name": "ohos.permission.INTERNET"
            }
        ]
    }
}
```

## 4.3 创建工作线程

```
getGlobalTaskDispatcher(TaskPriority.DEFAULT)
    .asyncDispatch(() -> doGet());
```

```
getGlobalTaskDispatcher(TaskPriority.DEFAULT)
    .asyncDispatch(() -> doPost());
```

## 4.4 Get请求

```

OkHttpClient client = new OkHttpClient();
Request request = new Builder()
    .url("http://192.168.0.128/t_student.json")
    .build();
Response response = null;

try {
    response = client.newCall(request).execute();
    String json = response.body().string();

    ZSONArray students = ZSONArray.stringToZSONArray(json);
    for (int i = 0; i < students.size(); ++i) {
        ZSONObject student = students.getZSONObject(i);

        int id = student.getInteger("id");
        String name = student.getString("name");
        int age = student.getInteger("age");
        int sex = student.getInteger("sex");
        String subject = student.getString("subject");

        HiLog.info(label, "%{public}d, %{public}s, " +
            "%{public}d, %{public}d, %{public}s",
            id, name, age, sex, subject);
    }
}
catch (IOException exception) {
    HiLog.info(label, exception.getLocalizedMessage());
}
finally {
    if (response != null)
        response.close();
}

```

## 4.5 Post请求

```

OkHttpClient client = new OkHttpClient();
FormBody form = new FormBody.Builder()
    .add("username", "tarena")
    .add("password", "123456")
    .build();
Request request = new Builder()
    .url("http://192.168.0.128/t_student.json")
    .post(form)
    .build();
Response response = null;

try {
    response = client.newCall(request).execute();
    String html = response.body().string();

    HiLog.info(label, html);
}
catch (IOException exception) {
    HiLog.info(label, exception.getLocalizedMessage());
}
finally {
    if (response != null)
        response.close();
}

```

## 例程：OkHttp

...\OkHttp\entry\build.gradle

```

...
dependencies {
    ...
    implementation 'com.squareup.okhttp3:okhttp:4.9.0'
}
...

```

...\OkHttp\entry\src\main\config.json

```
{
  ...
  "deviceConfig": {
    "default": {
      "network": {
        "cleartextTraffic": true
      }
    }
  },
  "module": {
    ...
    "reqPermissions": [
      {
        "name": "ohos.permission.INTERNET"
      }
    ]
  }
}
```

...\OkHttp\entry\src\main\resources\base\graphic\background\_button.xml

```
<?xml version="1.0" encoding="utf-8"?>
<shape
  xmlns:ohos="http://schemas.huawei.com/res/ohos"
  ohos:shape="oval">
  <solid ohos:color="#22b14c"/>
</shape>
```

...\OkHttp\entry\src\main\resources\base\layout\ability\_main.xml



```
<?xml version="1.0" encoding="utf-8"?>
<DirectionalLayout
  xmlns:ohos="http://schemas.huawei.com/res/ohos"
  ohos:height="match_parent"
  ohos:width="match_parent"
  ohos:alignment="center"
  ohos:orientation="vertical"
  ohos:background_element="#385723">

  <Button
    ohos:id="$+id:btnGet"
    ohos:height="96vp"
    ohos:width="96vp"
    ohos:background_element="$graphic:background_button"
    ohos:text="GET"
    ohos:text_size="24fp"
    ohos:text_color="#ffffff"
  />

  <Button
    ohos:id="$+id:btnPost"
    ohos:height="96vp"
    ohos:width="96vp"
    ohos:top_margin="24vp"
    ohos:background_element="$graphic:background_button"
    ohos:text="POST"
    ohos:text_size="24fp"
    ohos:text_color="#ffffff"
  />

</DirectionalLayout>
```

..\OkHttp\entry\src\main\java\com\minwei\okhttp\slice>MainAbilitySlice.java

```

public class MainAbilitySlice extends AbilitySlice {
    private static final HiLogLabel label = new HiLogLabel(
        HiLog.LOG_APP, 0x00101,
        MainAbilitySlice.class.getCanonicalName());

    @Override
    public void onStart(Intent intent) {
        ...
        findComponentById(ResourceTable.Id_btnGet)
            .setClickedListener(component -> onGet());
        findComponentById(ResourceTable.Id_btnPost)
            .setClickedListener(component -> onPost());
    }
    ...
    private void onGet() {
        getGlobalTaskDispatcher(TaskPriority.DEFAULT)
            .asyncDispatch(() -> doGet());
    }

    private void onPost() {
        getGlobalTaskDispatcher(TaskPriority.DEFAULT)
            .asyncDispatch(() -> doPost());
    }

    private void doGet() {
        OkHttpClient client = new OkHttpClient();
        Request request = new Builder()
            .url("http://192.168.0.128/t_student.json")
            .build();
        Response response = null;

        try {
            response = client.newCall(request).execute();
            String json = response.body().string();

            ZSONArray students = ZSONArray.stringToZSONArray(json);
            for (int i = 0; i < students.size(); ++i) {
                ZSONObject student = students.getZSONObject(i);

                int id = student.getInteger("id");
                String name = student.getString("name");
                int age = student.getInteger("age");
                int sex = student.getInteger("sex");
                String subject = student.getString("subject");

                HiLog.info(label, "%{public}d, %{public}s, " +
                    "%{public}d, %{public}d, %{public}s",
                    id, name, age, sex, subject);
            }
        }
        catch (IOException exception) {
            HiLog.info(label, exception.getMessage());
        }
        finally {

```

```

        if (response != null)
            response.close();
    }
}

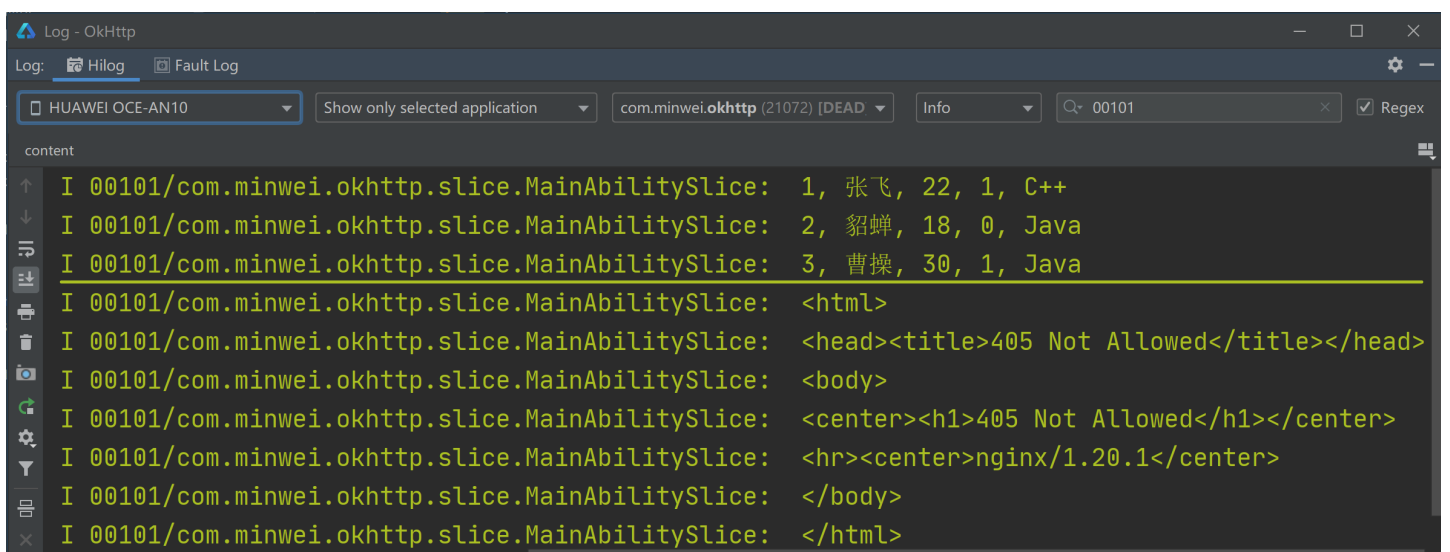
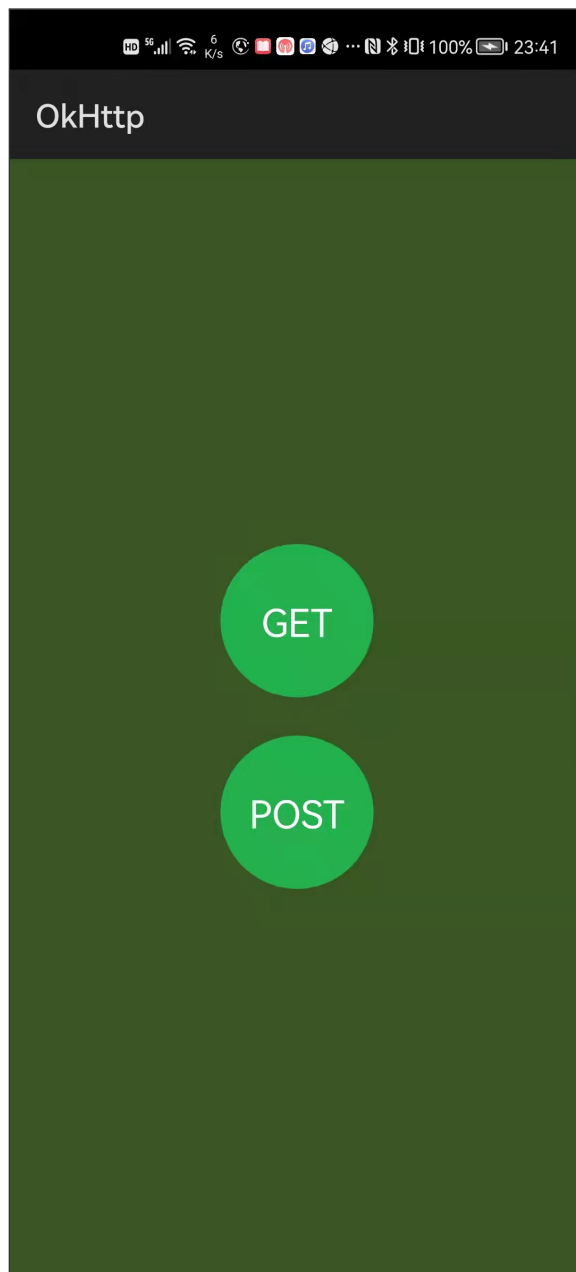
private void doPost() {
    OkHttpClient client = new OkHttpClient();
    FormBody form = new FormBody.Builder()
        .add("username", "tarena")
        .add("password", "123456")
        .build();
    Request request = new Builder()
        .url("http://192.168.0.128/t_student.json")
        .post(form)
        .build();
    Response response = null;

    try {
        response = client.newCall(request).execute();
        String html = response.body().string();

        HiLog.info(label, html);
    }
    catch (IOException exception) {
        HiLog.info(label, exception.getLocalizedMessage());
    }
    finally {
        if (response != null)
            response.close();
    }
}
}
}

```

运行效果如下图所示：



## 5 通过WebView组件访问Web服务器

### 5.1 添加权限并将Http设置为Web访问的缺省模式

config.json

```
{
  ...
  "deviceConfig": {
    "default": {
      "network": {
        "cleartextTraffic": true
      }
    }
  },
  "module": {
    ...
    "reqPermissions": [
      {
        "name": "ohos.permission.INTERNET"
      }
    ]
  }
}
```

## 5.2 WebView组件

```
<ohos.agp.components.webengine.WebView
  ohos:id="$+id:wv"
  ohos:height="match_parent"
  ohos:width="match_parent"/>
```

## 5.3 载入页面

```
wv.load("http://192.168.0.128");
```

例程：WebView

...\WebView\entry\src\main\config.json

```

{
  ...
  "deviceConfig": {
    "default": {
      "network": {
        "cleartextTraffic": true
      }
    }
  },
  "module": {
    ...
    "reqPermissions": [
      {
        "name": "ohos.permission.INTERNET"
      }
    ]
  }
}

```

...\WebView\entry\src\main\resources\base\layout\ability\_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<DirectionalLayout
  xmlns:ohos="http://schemas.huawei.com/res/ohos"
  ohos:height="match_parent"
  ohos:width="match_parent"
  ohos:alignment="center"
  ohos:orientation="vertical">

  <ohos.agp.components.webengine.WebView
    ohos:id="+id:wv"
    ohos:height="match_parent"
    ohos:width="match_parent"/>

</DirectionalLayout>

```

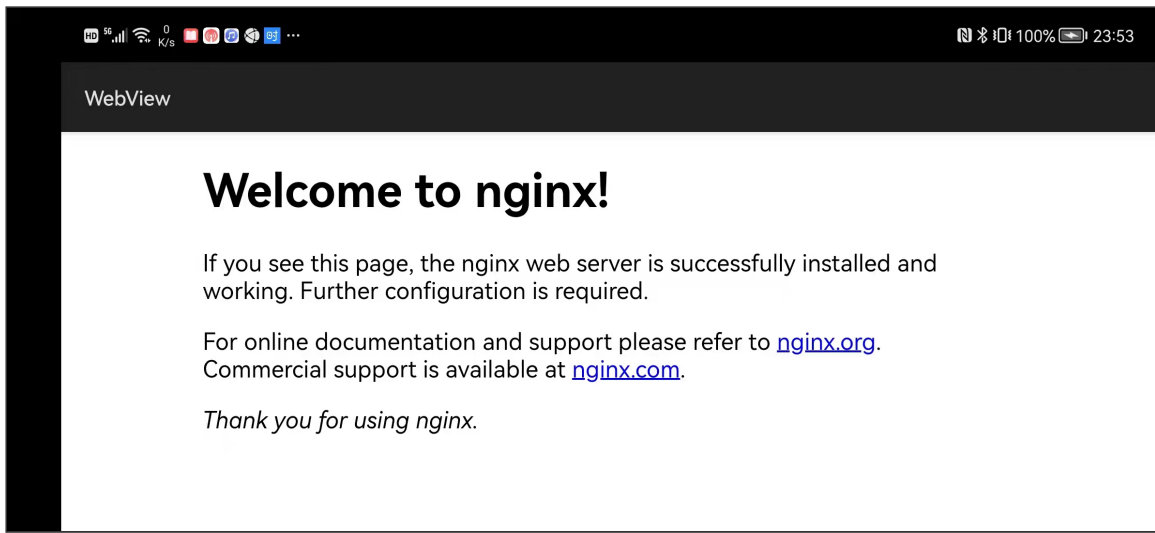
...\WebView\entry\src\main\java\com\minwei\webview\slice\MainAbilitySlice.java

```

public class MainAbilitySlice extends AbilitySlice {
  @Override
  public void onStart(Intent intent) {
    ...
    WebView wv = (WebView) findComponentById(ResourceTable.Id_wv);
    wv.load("http://192.168.0.128");
  }
  ...
}

```

运行效果如下图所示:



**更多精彩，敬请期待.....**

---

达内集团C++教学部  
2021年10月6日