



鸿蒙应用网络请求开发





前言

- 随着互联网的普及和技术的不断进步，网络管理的复杂性和重要性日益凸显。网络管理也是开发者在开发应用过程中经常会使用的重要功能。HarmonyOS中也统一存在网络管理模块提供相关能力。
- 本章将针对如何在HarmonyOS应用中进行网络管理，以及对其中使用度较高的HTTP数据请求技术进行详细讲解。



课程目标

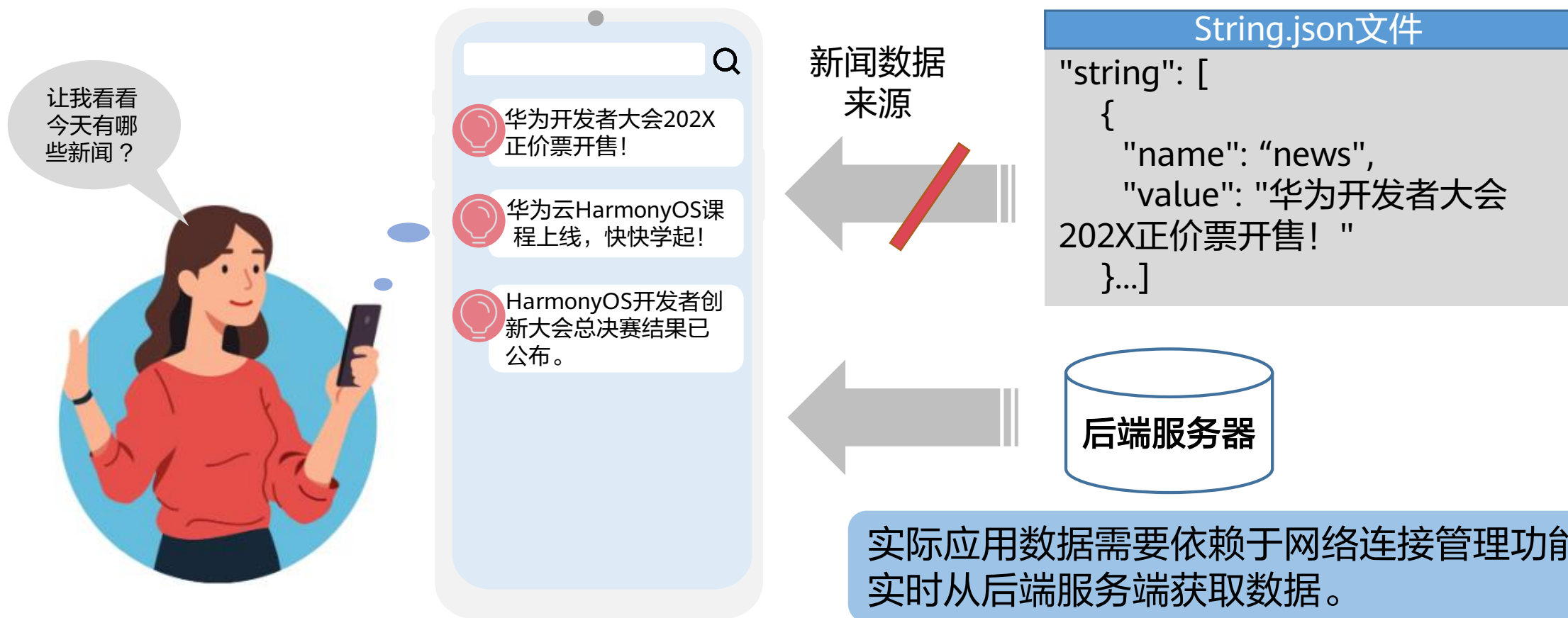
- 学完本课程后，您将能够：
 - 理解HarmonyOS中网络管理基本概念
 - 了解HTTP基本概念
 - 了解HarmonyOS中HTTP模块提供的各个接口功能
 - 掌握HTTP数据请求的具体开发步骤
 - 了解HTTP数据请求相关场景
 - 了解HarmonyOS中网络管理的约束与限制



1. HarmonyOS网络管理介绍
2. HarmonyOS中HTTP数据请求
3. 网络管理的约束与限制

应用数据从哪里来？

- 在前期实验中，我们构建UI页面时采用数据皆为静态数据。但实际的日常生活中应用数据的来源并不可能是静态数据。



网络管理开发概述

- HarmonyOS的网络管理模块主要提供以下三种网络连接功能：

★ HTTP数据请求：通过HTTP发起一个数据请求。

WebSocket连接：使用WebSocket建立服务器与客户端的双向连接。

Socket连接：通过Socket进行数据传输。

HTTP数据请求使用最为基础，应用较为广泛。因此，本章仅具体学习HTTP数据请求。



1. HarmonyOS网络管理介绍

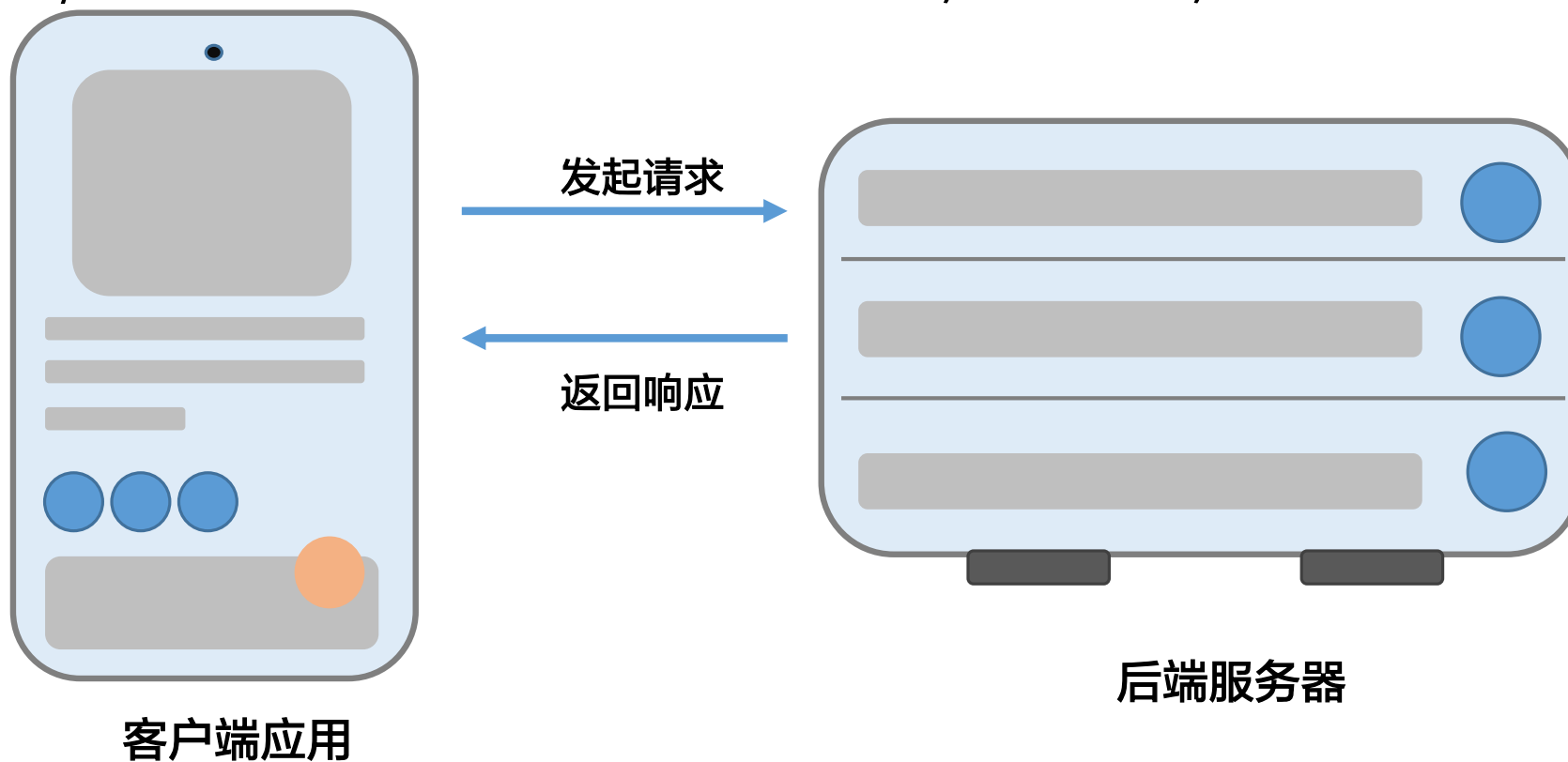
2. HarmonyOS中HTTP数据请求

- HTTP基本概念
- HTTP数据请求
- HTTP数据请求场景介绍

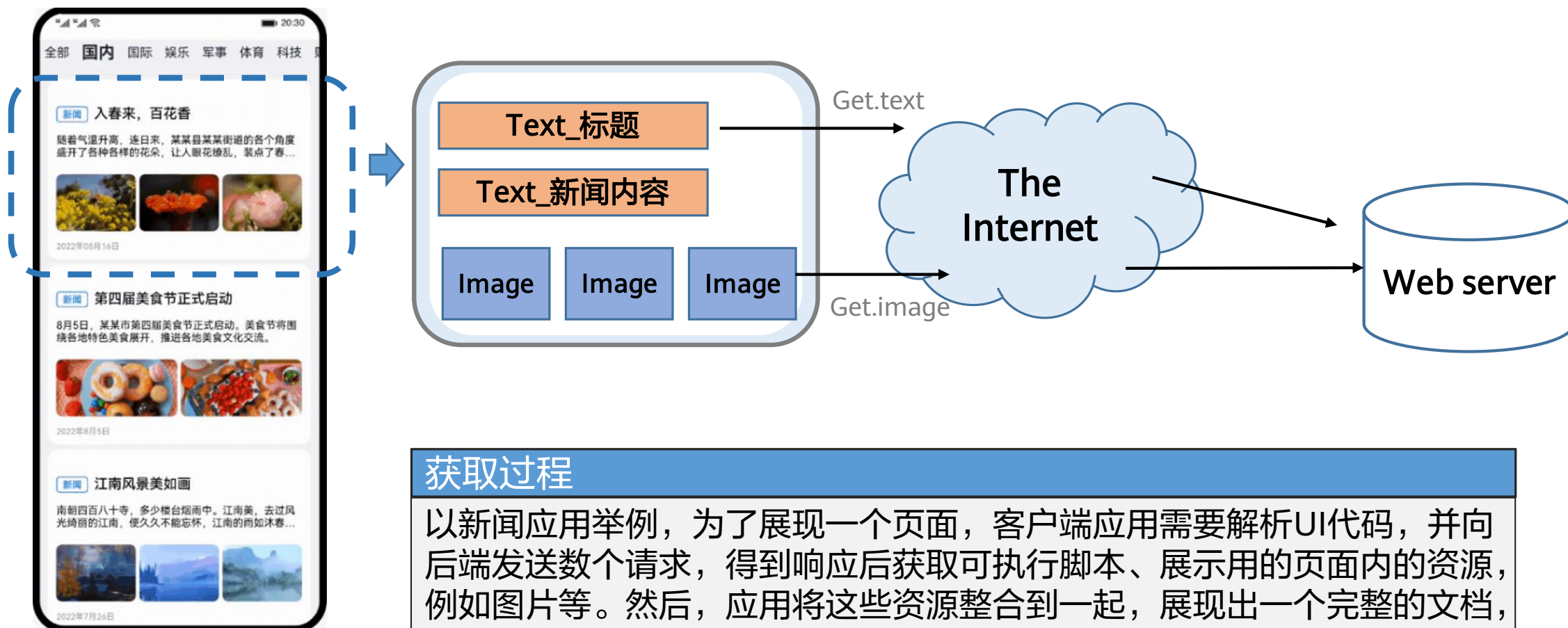
3. 网络管理的约束与限制

什么是HTTP?

- 超文本传输协议（Hyper Text Transfer Protocol, HTTP）是一个简单的请求响应协议。HTTP协议是在Web上进行数据交换的基础，也是一种“客户端-服务器端”协议。HTTP是一个基于TCP/IP通信协议来传递数据（HTML文件, 图片文件, 查询结果等）。



HTTP基本概念



获取过程

以新闻应用举例，为了展现一个页面，客户端应用需要解析UI代码，并向后端发送数个请求，得到响应后获取可执行脚本、展示用的页面内的资源，例如图片等。然后，应用将这些资源整合到一起，展现出一个完整的文档，即最左侧应用界面。



1. HarmonyOS网络管理介绍

2. HarmonyOS中HTTP数据请求

- HTTP基本概念
- HTTP数据请求
- HTTP数据请求场景介绍

3. 网络管理的约束与限制

HTTP数据请求

导入http模块

创建http请求

订阅请求头

发起请求

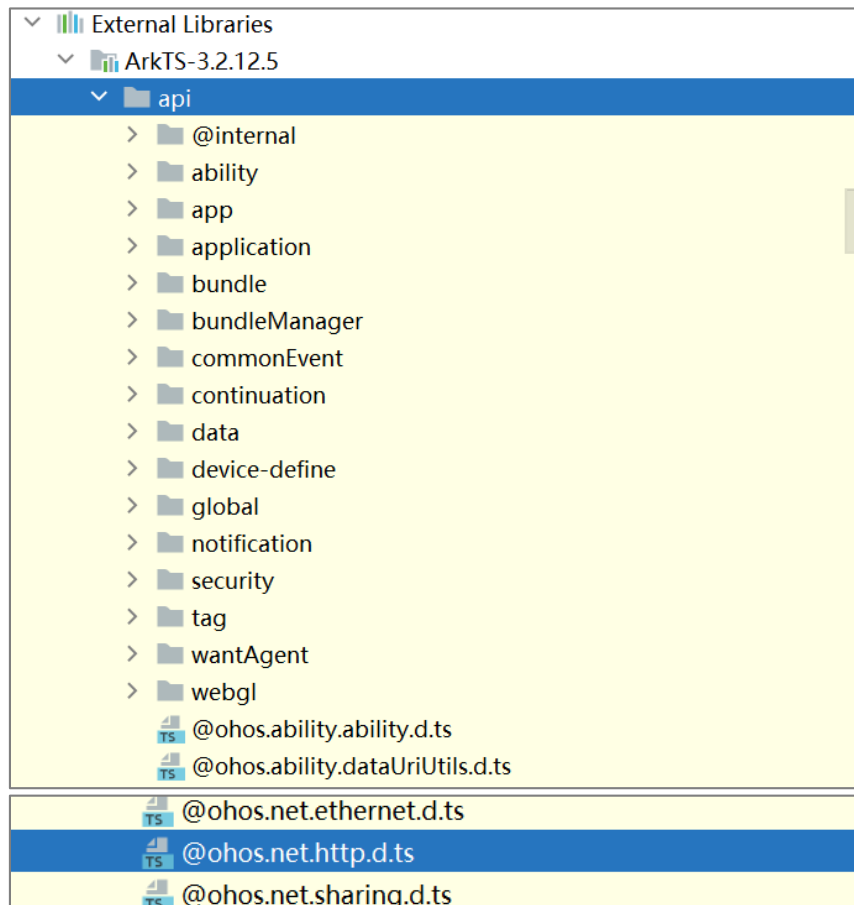
处理响应

从@ohos.net.http.d.ts中导入http命名空间。

```
// 引入包名
```

```
import http from '@ohos.net.http';
```

HarmonyOS工程文件中的External Libraries文件夹下罗列了该项目所依赖的资源库。这里我们所导入的HTTP模块就存放在该目录下的@ohos.net.http.d.ts文件中。



HTTP数据请求常用接口

- HTTP数据请求功能涉及的接口如下表：

接口名	功能描述
createHttp()	创建一个http请求。
request()	根据URL地址，发起HTTP网络请求。
destroy()	中断请求任务。
on(type: 'headersReceive')	订阅HTTP Response Header 事件。
off(type: 'headersReceive')	取消订阅HTTP Response Header 事件。
once('headersReceive')	订阅HTTP Response Header 事件，但是只触发一次。

HTTP数据请求

导入http模块

创建http请求

订阅请求头

发起请求

处理响应

从@ohos.net.http.d.ts中导入http命名空间。

```
// 引入包名
```

```
import http from '@ohos.net.http';
```

```
// 创建数据请求对象
```

```
// 每一个HttpRequest对应一个HTTP请求任务，不可复用
```

```
let httpRequest = http.createHttp();
```

creatHttp成功后，返回一个httpRequest对象，里面包括request、destroy、on和off方法。

HTTP数据请求

导入http模块

创建http请求

订阅请求头

发起请求

处理响应

调用该对象的on()方法，订阅http响应头事件，此接口会比request请求先返回。可以根据业务需要订阅此消息。

```
// 订阅请求头（可选步骤）  
// 回调函数中获取header的信息  
httpRequest.on('headersReceive', (header) => {  
  // 用JSON.stringify(header)解析头信息  
  console.info('header: ' + JSON.stringify(header));  
});
```

用于订阅HTTP响应头，订阅事件类型：'headerReceive'，可以根据业务需要订阅此信息。

HTTP数据请求

导入http模块

创建http请求

订阅请求头

发起请求

处理响应

调用该对象的request()方法，传入http请求的url地址和可选参数，发起网络请求。

```
httpRequest.request(  
// 填写HTTP请求的URL地址。URL地址需要开发者自定义。请求的参数可以在extraData中指定。  
"EXAMPLE_URL",  
{  
method: http.RequestMethod.POST, // 请求方式，默认为http.RequestMethod.GET  
header: {'Content-Type': 'application/json'}, // 开发者根据自身业务需要添加header字段（HTTP请求头字段）  
extraData: {  
"data": "data to send",  
}, // extraData发送请求的额外数据  
readTimeout: 60000, // 可选，读取超时事件，默认为60000ms  
},
```

HTTP数据请求

导入http模块

创建http请求

订阅请求头

发起请求

处理响应

调用该对象的request()方法，传入http请求的url地址和可选参数，发起网络请求。

```
httpRequest.request(  
// 填写HTTP请求的URL地址。URL地址需要开发者自定义。请求的参数可以在extraData中指定。
```

```
"EXAMPLE_URL",
```

1、请求URL路径

```
{  
method: http.RequestMethod.POST,  
header: {'Content-Type': 'application/json'},  
extraData: {  
"data": "data to send",  
},  
readTimeout: 60000,  
},
```

2、请求选项 HttpRequestOptions

HTTP数据请求

导入http模块

创建http请求

订阅请求头

发起请求

处理响应

HttpRequestOptions

发起请求可选参数的类型和取值范围。

名称	类型	描述
method	RequestMethod	请求方式, GET、POST、PUT、DELETE
extraData	string Object	请求参数
header	Object	请求头字段
connectTimeout	number	连接超时时间。单位为毫秒 (ms), 默认为60000ms。
readTimeout	number	读取超时时间。单位为毫秒 (ms), 默认为60000ms。 设置为0表示不会出现超时情况。

HTTP数据请求

导入http模块

创建http请求

订阅请求头

发起请求

处理响应

按照实际业务需要，解析返回结果。

```
// 发送数据请求，详见上文。
```

```
httpRequest.request(  
.....  
},
```

```
// 服务端返回结果
```

```
Promise.then( ((value) =>{
```

```
    if(value.responseCode === 200){
```

```
        // 处理响应 value为请求返回结果
```

```
        // 包含value.responseCode、value.Result
```

```
        // value.header、value.cookies四个字段
```

```
    }
```

```
})
```

HttpResponse

名称	类型	描述
responseCode	ResponseCode	响应状态码
header	Object	发起HTTP请求返回来的响应头
cookies	String	响应返回的 cookies
result	String Object	响应体，默认为JSON字符串
resultType	HttpDataType	返回值类型

ResponseCode状态码对应表格

- 请求返回的响应码如下图，详细响应码可以查询API文档。
- 调用时可以使用：`http. ResponseCode.OK`，来表示数值为200，请求成功。

名称	值	说明
OK	200	请求成功。一般用于GET与POST请求。
CREATED	201	已创建。成功请求并创建了新的资源。
ACCEPTED	202	已接受。已经接受请求，但未处理完成。
NOT_AUTHORITY	203	非授权信息。请求成功。
NO_CONTENT	204	无内容。服务器成功处理，但未返回内容。
RESET	205	重置内容。
...

试一试：

- 在第八章中，我们将会学习如何使用HTTP请求调用华为云平台的FunctionGraph云函数并创建自己的URL进行开发。这里我们暂时填写HTTP请求的URL测试地址为并打印其返回响应数据：

```
"https://71bd6fb409224c4298f7d7dc103414b7.apig.cn-north-4.huaweicloudapis.com/user/get",
```

```
// data.result为HTTP响应内容，可根据业务需要进行解析  
console.info('Result:' + JSON.stringify(data.result));  
console.info('code:' + JSON.stringify(data.responseCode));  
// data.header为HTTP响应头，可根据业务需要进行解析  
console.info('header:' + JSON.stringify(data.header));  
console.info('cookies:' + JSON.stringify(data.cookies)); // 8+
```

```
app Log: header: {"connection":"keep-alive","content-length":"43","content-type":"application  
app Log: Result:"[{"name":"张三","age":"18","class":"10"}]"  
app Log: code:200  
app Log: header:{"connection":"keep-alive","content-length":"43","content-type":"application  
app Log: cookies:""
```

日志打印结果



1. HarmonyOS网络管理介绍

2. HarmonyOS中HTTP数据请求

- HTTP基本概念
- HTTP数据请求
- HTTP数据请求方式介绍

3. 网络管理的约束与限制

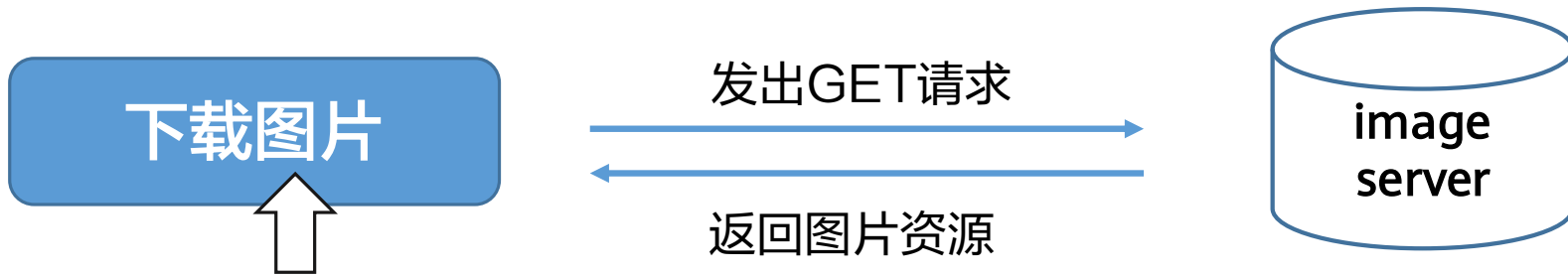
HTTP支持的请求方式

- 应用可以通过HTTP发起一个数据请求，支持常见的GET、POST、OPTIONS、HEAD、PUT、DELETE、TRACE、CONNECT方法。其中GET、POST请求在实际应用场景中使用较多。

序号	字段	使用场景
1	GET	请求指定的页面信息，并返回响应主体
2	POST	请求会向指定资源提交数据，请求服务器进行处理
3	PUT	请求会向指定资源位置上传其最新内容
4	CONNECT	HTTP/1.1协议预留的，能够将连接改为管道方式的代理服务器
5	HEAD	类似于GET请求，返回响应头信息，但是不会返回响应主体
6	DELETE	请求用于请求服务器删除所请求URI
7	TRACE	请求服务器回显其收到的请求信息
8	OPTIONS	请求用于客户端查看服务器的性能

HTTP支持的请求方式：GET请求举例

- 用于请求服务器发送某个资源。GET 请求不应该对服务器上的资源做出任何更改，并且应该是幂等的（即多次重复的请求应该产生相同的结果）。
- 相当于数据库中的查询动作。
- GET常见应用场景：
 1. 登录时GET获取服务器数据库用户名和密码进行验证。
 2. 下载文本、图片、音视频等时获取服务器资源。



HTTP支持的请求方式: GET

```
let url = "https://EXAMPLE_URL?param1=v1&param2=v2";  
let promise = httpRequest.request(url,{  
  methd:http.requestMethond.GET,  
  connecTimeout:60000,  
  readTimeout:60000,  
  header:{  
    'Content-Type':'application/json'  
  }  
});
```

GET请求
参数内容会拼接到URL中

HTTP支持的请求方式：POST请求

- POST请求会对指定资源提交数据进行处理请求。数据被包含在POST请求体中。例如上文发起请求案例中，使用extraData字段包含了需要传递的内容，而不是在URL中进行拼接。
- POST 请求常见应用场景：
 1. 提交用户注册信息。
 - 例如，在提交注册表单时，应用通常会发送一个 POST 请求，将用户提供的信息发送到服务器进行处理。
 2. 提交修改的用户信息。

目录

1. HarmonyOS网络管理介绍
2. HarmonyOS中HTTP数据请求
- 3. 网络管理的约束与限制**

权限声明

- HarmonyOS根据接口所涉数据的敏感程度或所涉能力的安全威胁影响，定义了不同开放范围与授权方式的**权限**来保护数据。当应用调用服务时，服务会对应用进行权限检查，如果**没有对应权限则无法使用该服务**。
- 应用在使用对应服务的能力或数据时，需要申请对应权限。
 - 已在**module.json5配置文件**文件中声明的非敏感权限，会在应用安装时自动授予，该类权限的授权方式为系统授权 (system_grant)。
 - 敏感权限需要应用动态申请，通过**运行时发送弹窗**的方式请求用户授权，该类权限的授权方式为用户授权 (user_grant)。

网络管理约束与限制

- HTTP数据请求功能主要由http模块提供，需要允许应用可以使用Internet网络。
- 所以在使用该功能需要必须优先申请`ohos.permission.INTERNET`权限，该权限为非敏感权限。

```
// 在module.json5配置文件中添加网络管理权限
"module": {
  "requestPermissions": [
    {"name": "ohos.permission.INTERNET"}
  ]
}
```



本章小结

- 主要通过HarmonyOS网络请求相关库，介绍了HTTP数据请求的相关API使用方式。通过本章的学习，可以掌握应用开发中网络请求与连接功能的基础开发能力，并可以为进一步学习更多功能开发方式打下良好基础。

思考题

1. （单选题）开发者想要使用HTTP数据请求功能，就需要先开启Internet网络权限。需要以下哪一个配置文件中设置权限？（ ）
- A. module.json5
 - B. app.json5
 - C. build-profile.json5
 - D. hvigorfile.ts

思考题

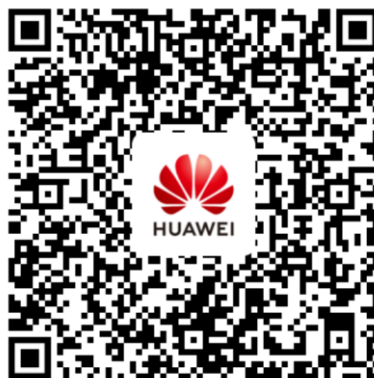
2. (判断题) GET请求会对指定资源提交数据进行处理请求。数据在URL中进行拼接。()
- A. 正确
 - B. 错误



学习推荐

- 官方学习网站

- HarmonyOS官网: <https://developer.harmonyos.com/>
- HarmonyOS应用开发文档: <https://developer.huawei.com/consumer/cn/>
- OpenHarmony官网: <https://edu.huaweicloud.com/>
- 华为开发者论坛: <https://developer.huawei.com/consumer/cn/forum/>



华为云开发者学堂

感谢

版权所有©2024，华为技术有限公司，保留所有权利。

本资料是华为的保密信息，所有内容仅供华为授权的培训客户内部使用，禁止用于任何其他用途。未经许可，任何人不得对本资料进行复制、修改、改编、也不得将本资料或其任何部分或基于本资料的衍生作品提供给他人。

