REST（英文：Representational State Transfer，简称REST）

Web 应用程序最重要的 REST 原则是，客户端和服务器之间的交互在请求之间是**无状态**的。从客户端到服务器的每个请求都必须包含理解请求所必需的信息。

**REST 指的是一组架构约束条件和原则。满足这些约束条件和原则的应用程序或设计就是 RESTful。**

在服务器端，应用程序状态和功能可以分为各种**资源**。资源是一个有趣的概念实体，它向客户端公开。资源的例子有：应用程序对象、数据库记录、算法等等。每个资源都使用 URI (Universal Resource Identifier) 得到一个唯一的地址。所有资源都共享统一的接口，以便在客户端和服务器之间传输状态。**使用的是标准的 HTTP 方法，比如 GET、PUT、POST 和 DELETE。**

（1）每一个**URI**代表一种资源；

（2）客户端通过四个HTTP动词，对服务器端资源进行操作，实现"表现层状态转化"。

另一个重要的 REST 原则是分层系统，这表示组件无法了解它与之交互的中间层以外的组件。

GET /zoos：列出所有动物园

POST /zoos：新建一个动物园

GET /zoos/ID：获取某个指定动物园的信息

PUT /zoos/ID：更新某个指定动物园的信息（提供该动物园的全部信息）

PATCH /zoos/ID：更新某个指定动物园的信息（提供该动物园的部分信息）

DELETE /zoos/ID：删除某个动物园

GET /zoos/ID/animals：列出某个指定动物园的所有动物

DELETE /zoos/ID/animals/ID：删除某个指定动物园的指定动物

基于http，更规范，更适合对外开放的api