

**RESTful API**

**Virtual  
Internship  
Experience**

  
**mandiri**

# Disclaimer

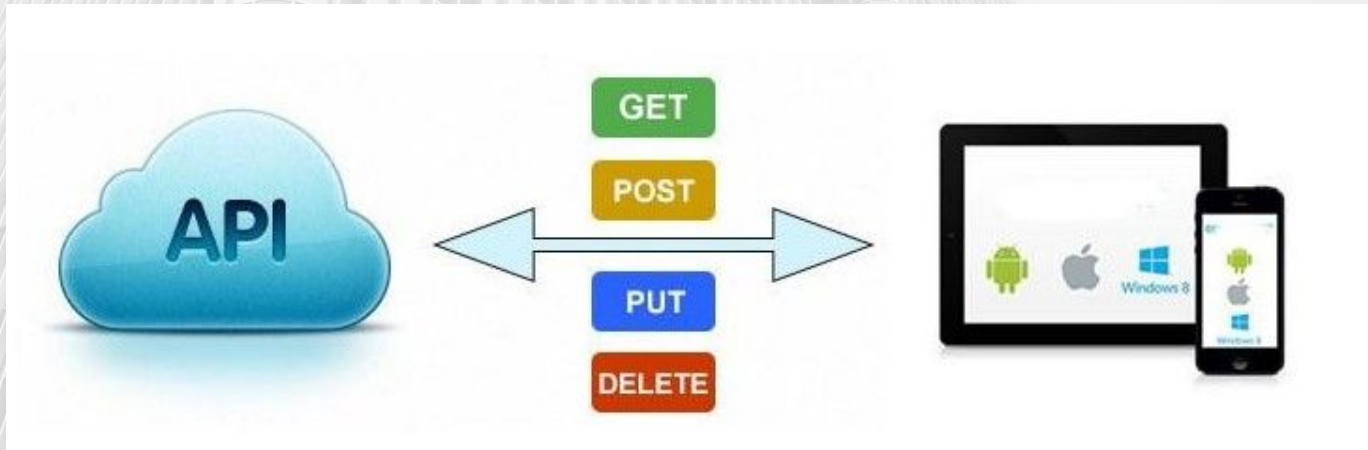
“Dokumen ini memiliki hak cipta. Barang siapa yang menyebarluaskan atau menduplikasi tanpa izin dari instansi terkait dapat diproses sesuai dengan ketentuan hukum yang berlaku.”

# Outline

- 1 RESTful API
- 2 Case Study

# RESTful API

- RESTful API merupakan penerapan dari API (*Application Programming Interface*).
- Bertujuan untuk menjadikan sistem memiliki performa yang baik, cepat dan mudah untuk di kembangkan (*scale*) terutama dalam pertukaran dan komunikasi data.





# Komponen RESTful API



1. **URL Design** : RESTful API diakses menggunakan protokol HTTP. URL API biasa disebut *endpoint* dalam pemanggilannya.
2. **HTTP Verbs** : Setiap *request* yang dilakukan terdapat metode yang dipakai agar server mengerti apa yang sedang di-*request client*. Metode yang ada GET, POST, PUSH, DELETE, dan sebagainya.
3. **HTTP Response Code** : Kode standarisasi dalam menginformasikan hasil *request* kepada *client*. Secara umum terdapat 3 kelompok yang biasa dijumpai pada RESTful API yaitu :
  - 2XX : adalah *response code* yang menampilkan bahwa *request* berhasil.
  - 4XX : adalah *response code* yang menampilkan bahwa *request* mengalami kesalahan pada sisi *client*.
  - 5XX : adalah *response code* yang menampilkan bahwa *request* mengalami kesalahan pada sisi server.
4. **Format Response** : Setiap *request* yang dilakukan *client* akan menerima data *response* dari server, *response* tersebut biasanya berupa data XML ataupun JSON.

# Case Study

# Membuat Aplikasi Superhero menggunakan Retrofit part II

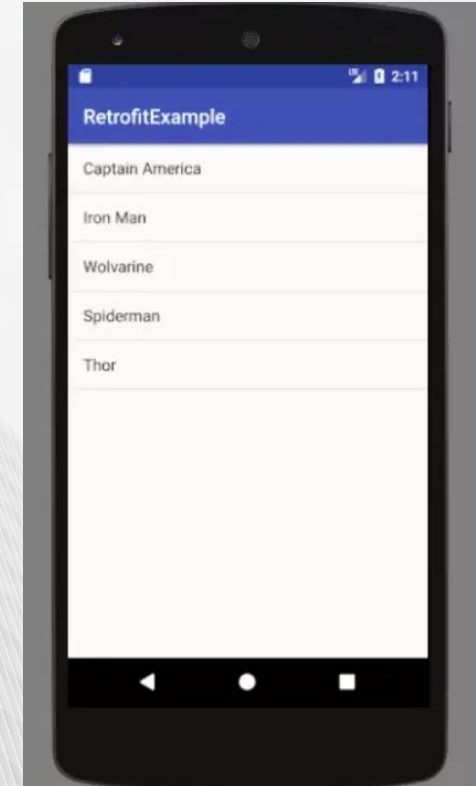


## Review Case Study pada Video Learning 7

API : <https://simplifiedcoding.net/demos/marvel/>

Response :

```
▼ 0:
  name:      "Captain America"
  realname:   "Steve Rogers"
  team:      "Avengers"
  firstappearance: "1941"
  createdby:  "Joe Simon"
  publisher:  "Marvel Comics"
  ▼ imageUrl: "https://www.simplifiedcoding.net/demos/marvel/captainamerica.jpg"
  ▼ bio:      "\r\n\t\tSteven Rogers was born in the Lower East Side of Manhattan, New York City, in 1925
to poor Irish immigrants, Sarah and Joseph Rogers.[54] Joseph died when Steve was a child,
and Sarah died of pneumonia while Steve was a teen. By early 1940, before America's entry
into World War II, Rogers is a tall, scrawny fine arts student specializing in illustration
and a comic book writer and artist.\r\n\n\t\t"
```



# Membuat Aplikasi Superhero menggunakan Retrofit part II



**Step #6 :** Buat Interface untuk endpoint marvel

```
import retrofit2.Call
import retrofit2.http.GET

interface ApiInterface {
    @GET("marvel")
    fun getHeroes(): Call<List<Heroes?>>?
}
```

<https://simplifiedcoding.net/demos/marvel/>

**Base URL**

**Path  
Endpoint**



# Membuat Aplikasi Superhero menggunakan Retrofit part II



**Step #7** : Buat file dengan nama item\_hero.xml. Kemudian buat layout UI untuk menampilkan data superhero pada RecyclerView nantinya.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <TextView
        android:id="@+id/tv_name"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

</LinearLayout>
```

# Membuat Aplikasi Superhero menggunakan Retrofit part II



**Step #8** : Buat class HeroesAdapter untuk memuat list data superhero pada RecyclerView

```
class HeroesAdapter(private val heroes : ArrayList<Heroes>) :  
    RecyclerView.Adapter<HeroesAdapter.HeroViewHolder>() {  
  
    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): HeroViewHolder {  
        val inflater = LayoutInflater.from(parent.context)  
        val binding = ItemHeroBinding.inflate(inflater)  
        return HeroViewHolder(binding)  
    }  
  
    override fun onBindViewHolder(holder: HeroViewHolder, position: Int) {  
        holder.bind(heroes[position])  
    }  
  
    override fun getItemCount() = heroes.size  
  
    class HeroViewHolder(val binding : ItemHeroBinding) : RecyclerView.ViewHolder(binding.root) {  
        fun bind(hero: Heroes) {  
            binding.tvName.text = hero.name  
        }  
    }  
}
```

# Membuat Aplikasi Superhero menggunakan Retrofit part II



**Step #9** : Inisialisasi class ApiClient dan HIT API marvel pada class MainActivity pada method onCreate()

```
val apiInterface : ApiInterface = ApiClient.getClient().create(ApiInterface::class.java)
val call : Call<Heroes> = apiInterface.getHeroes()
    call.enqueue(object : Callback<Heroes> {
        override fun onFailure(call: Call<Heroes>?, t: Throwable?) {
            Log.d("HIT_HEROES", "Gagal Fetch Data Superhero")
        }

        override fun onResponse(call: Call<Heroes>?, response: Response<Heroes>?) {
            heroes= response!!.body()!!.results
            Log.d("HIT_HEROES", "Heroes size ${heroes.size}")
            binding.rvHeroes.adapter = HeroesAdapter(heroes)
        }
    })
})
```



# Thank You!



mandiri