Global Future Cities Programme

Smart GIS Training: Session 1
GIS Fundamentals

26 October 2021 (15:30)

Agenda

1. Overview
2. GIS Fundamentals
3. Application
4. Practical

Introductions

Jonathan McCallum
Senior Consultant

Nurul Sofiana Syafril
GIS Engineer

Matthew Fredericks
Data Scientist

Overview

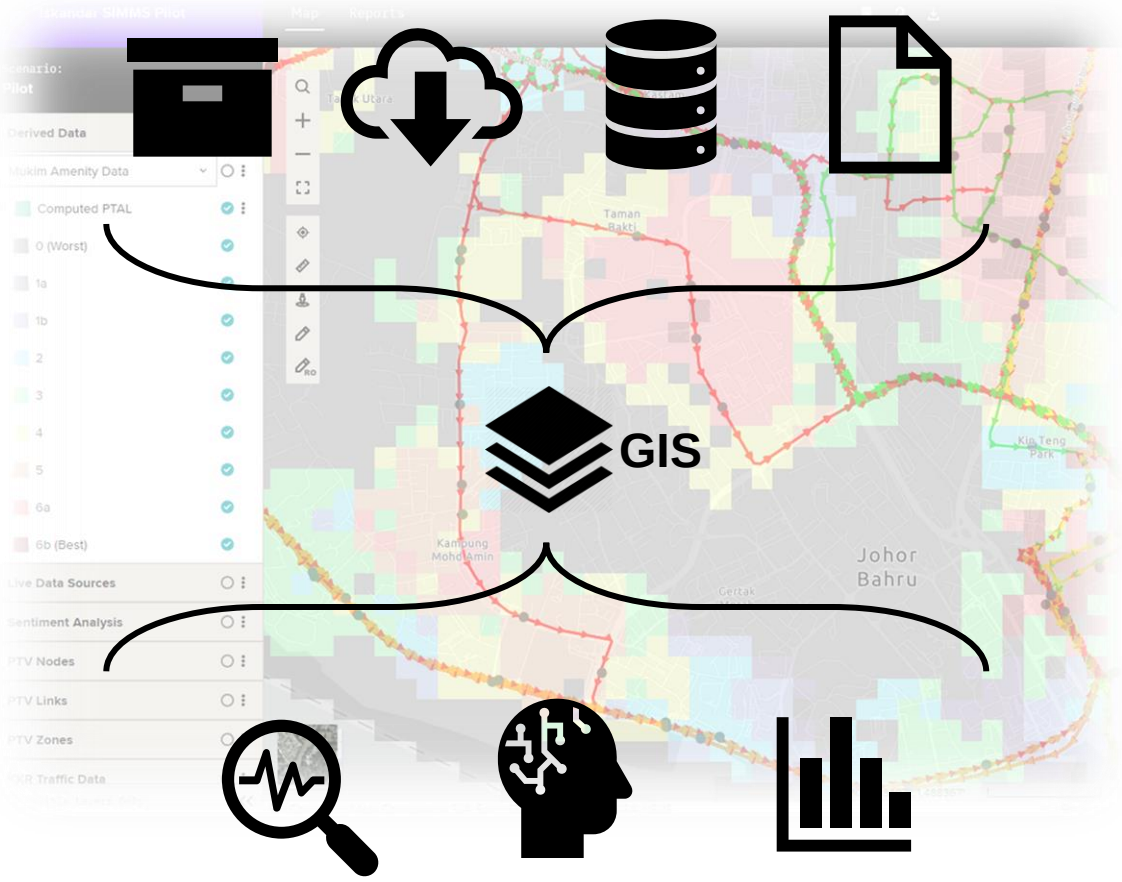
Overview

Over the next 3 sessions we will look at the work performed to produce the Smart GIS as part of the pilot project on the Iskandar intervention:

- Collect Data
- Process it into GIS formats
- Apply analytics
- Produce visualisations
- Generate additional functionality

We will cover:

- GIS Fundamentals
- Derivation of Data
- Advanced Analytics



FUTURE CITIES
Global Future Cities Programme

GIS Fundamentals

GIS Fundamentals

Basic GIS data introduction

GIS Best Practice

- Naming
- Fields & Attribute

Online GIS Introduction

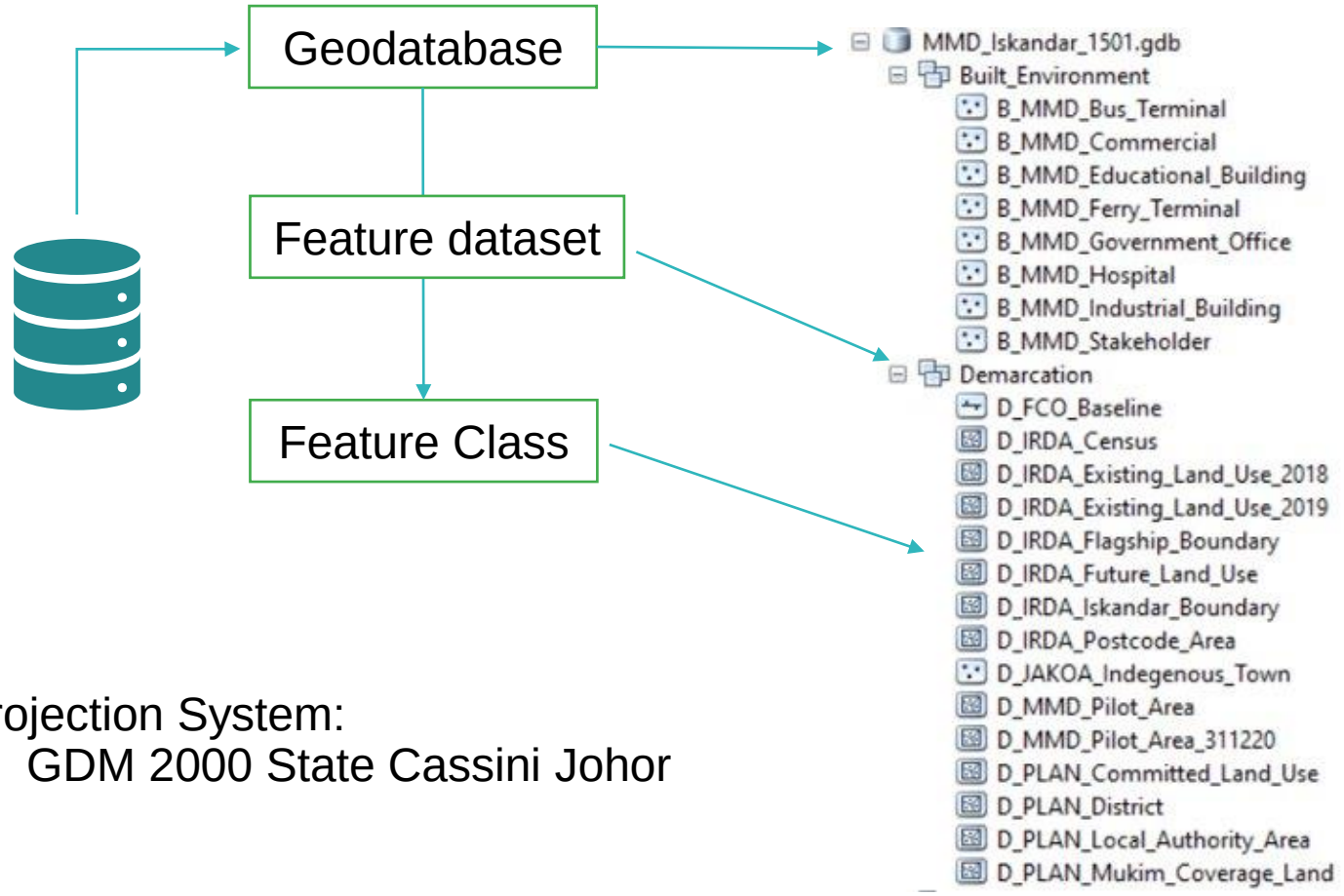
- Data formats
- Data template preparation

Cloud storage, direct links, API links

How do we use APIs?

Basics

SHAPEFILE	Category	GEODATABASE
• Groups several individual files that work together (typically geometry, attributes, and indexing) • Near-universal compatibility with all GIS software packages • Considered the most portable format in GIS as users outside of Esri environments/geodatabases can access files	Key Benefits	• Groups similar data into one database • Allows topology setup, coded value domains, raster catalogs, relationship classes, and geometric networks • Increases spatial function performance speed
Greater than .gdb	File Size	Less than .shp
10 characters	Field Name Length Limit	64 characters
2 GB	File Size Limit	Limited by computer hard drive storage space
Limited only by operating system	Table /Feature Class Name Length	160 characters
Vector	Data Types	Vector, Raster and more complex data such as topology
Can't be used in a field (0 used instead)	NULL Values	Supported
No	File Compression	Yes
No	Proprietary Esri Format	Yes



Projection System:

- GDM 2000 State Cassini Johor

Best Practice - Standard

- THM – or theme. This is a 1-letter abbreviation indicating the dataset theme or category in accordance with MS1759:2015 .
- ORIG – or Originator. This is a 3 to 5 letter abbreviation depicting the name of company who first created the dataset.
- BriefName – A short descriptive name describing the content of the dataset. This should not have spaces, separate words with an underscore is used instead.
- Suffix – To modify or add descriptive text to a filename.

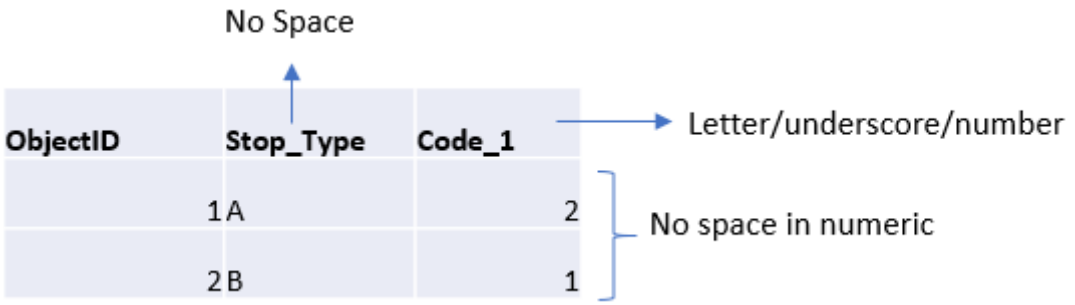
Theme	Originator Code	Feature Name	Suffix	Feature Name
B	MMD	LastBusTerminal	Pt	B_MMD_LastBusTerminal
B	MMD	Stakeholder	Pt	B_MMD_Stakeholder
T	IRDA	CCTV	Pt	T_IRDA_CCTV

Processing

- Data Standards
- Field type and properties
- Field names
- Field characters
- Attribute Accuracy

Field type	Storable range	Size (Bytes)	Applications
Short integer	-32,768 to 32,767	2	Numeric values without fractional values within specific range; coded values
Long integer	-2,147,483,648 to 2,147,483,647	4	Numeric values without fractional values within specific range
Float (single-precision floating-point number)	approximately -3.4E38 to 1.2E38	4	Numeric values with fractional values within specific range
Double (double-precision floating-point number)	approximately -2.2E308 to 1.8E308	8	Numeric values with fractional values within specific range

Feature Class	Subtype	Domain
		Officer A
	New	Officer B
	Completed	None
		Officer C
	Investigate	Officer D

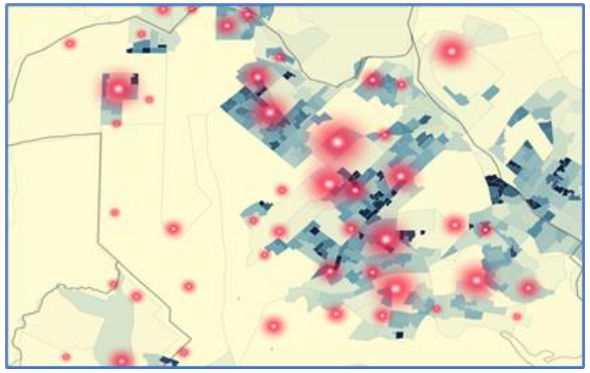


Only use text and numeric characters; avoid using special characters such as “! @ # % { } | / \ ~”. Commas should not be used in fields as they may cause problems when transferring data. Separations should either be represented as a space, underscore or a hyphen.

Online GIS & Web Layer

Hosted feature layer is a group of similar geographic features (point, line, polygon) that have been published (stored) in ArcGIS Online

- GeoJSON file (.geojson or .json)
- Microsoft Excel files (.xlsx or .xls)
- Comma-separated values (CSV) file (.csv)
- File geodatabase (.zip)
- Shapefile (.zip)
- etc



Web Map

Type of other web layers:

- Imagery layer
- Map Image layer
- Tile layer
- Scene layer

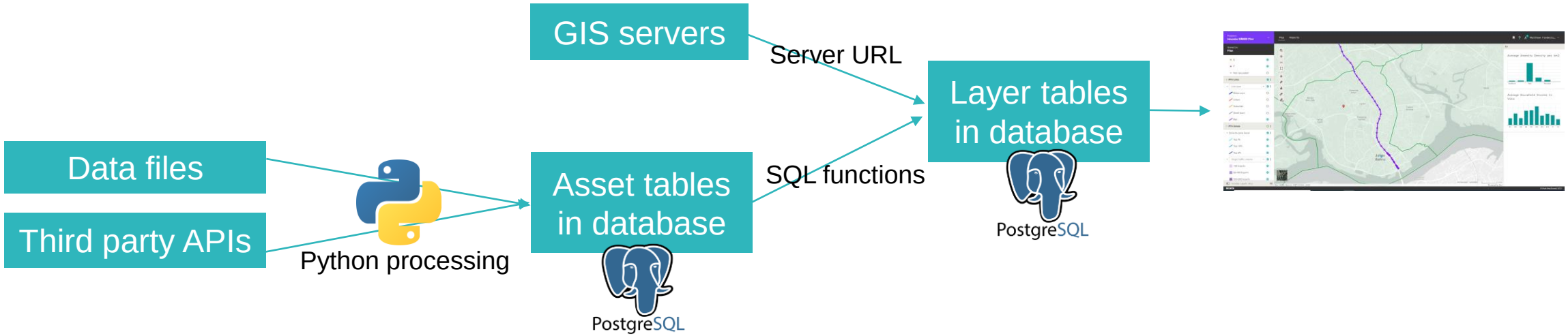
Layers:

- T_FCO_Road (0)
- T_MMD_Weighting_Station_Complex (1)
- T_PAJ_Bus_Outsite_Iskandar (2)
- T_PAJ_Bus_Within_Iskandar (3)
- T_CCTV (4)
- B_MMD_Commercial (5)
- B_MMD_Educational_Building (6)
- B_MMD_Industrial_Building (7)
- B_MMD_Hospital (8)
- B_MMD_Government_Office (9)
- B_MMD_Ferry_Terminal (10)
- B_MMD_Bus_Terminal (11)
- B_MMD_Stakeholder (12)
- D_PLAN_District (13)
- D_PLAN_Mukim_Coverage_Land (14)
- D_PLAN_Local_Authority_Area (15)
- D_MMD_Pilot_Area (16)
- D_IRDA_Existing_Land_Use_2019 (17)
- D_PLAN_Committed_Land_Use (18)
- D_FCO_Baseline (19)
- D_IRDA_Census (20)
- D_IRDA_Postcode_Area (21)
- D_IRDA_Iskandar_Boundary (22)
- D_IRDA_Future_Land_Use (23)
- D_IRDA_Flagship_Boundary (24)
- D_IRDA_Existing_Land_Use_2018 (25)
- D_JAKOA_Indegenous_Town (26)
- U_MMD_Meteorological_Station (27)
- U_MMD_Rainfall_Station (28)
- U_IRDA_ChargeEV_Unit (29)
- U_IRDA_ChargeEV_Station (30)

Cloud storage, direct links, API links

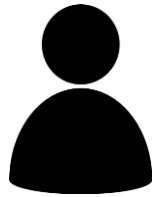
Overview

- Objective: display all kinds of geospatial data in one place
- ArcGIS servers, data files such as .shp and .gdb, third party APIs
- Solution: Moata platform



Find out more about Moata's applications: https://www.mottmac.com/digital/moata

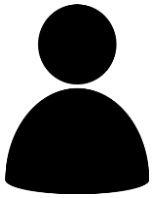
How do we use APIs?



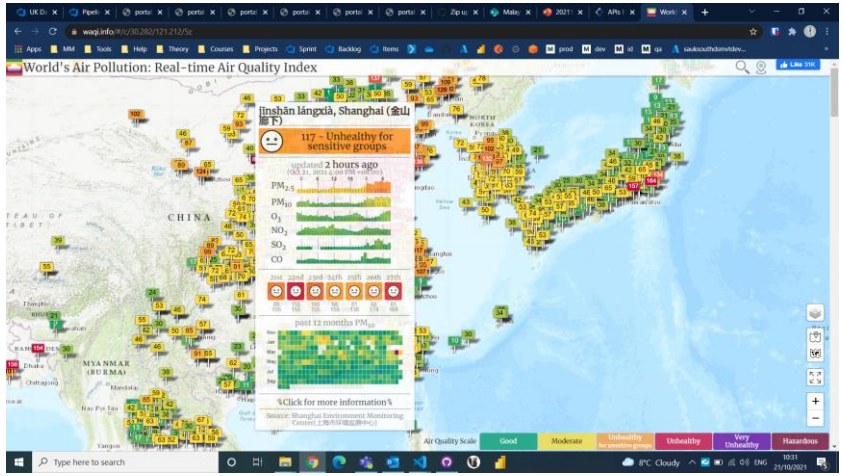
API

- API: Application Programming Interface

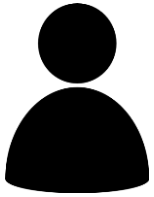
I want to know the air quality in Shanghai



Manual




```
import requests
url = 'https://api.waqi.info/feed/shanghai/?token=...'
req = requests.get(url)
data = req.content
```



API

```
{
   status: "ok",
   data: {
      aqi: 70,
      time: {
         s: "2021-06-20 09:00:00"
      },
      city: {
         name: "Shanghai",
         url: "http://aqicn.org/city/shanghai/",
         geo: [
            "31.2047372",
            "121.4489017"
         ]
      },
      iaqi: {
         pm25: "..."
      }
   }
}
```

- A programmatic API will allow you to get data from a website/app in an **automated** way

Application

Application

ArcGIS servers

- Each layer must point towards a data source to visualise
- Given a url we can extract the necessary info (geometries, attributes, styling)
  - For example: https://onemap.mbip.gov.my/arcgis/rest/services/.../MapServer
  - Other supported types include FeatureServer urls and Web Map Service/Web Map Tile Service urls

Interface to allow users to add their own layers is currently in testing

Application

Data files

- Files are processed in Python to tabular format with text-represented geometries

	A	B	D	F	
1	unique_id	route_id	num_trips	route_long_name	geometry
2	Route_0321cfcbecb7f6e062f1d294_inbound	0321cfcbecb7f6e062f1d294	9	Service ID CT1	LINESTRING (103.78926 1.49886, 103.78899 1.49863, 103.
3	Route_0321cfcbecb7f6e062f1d294_outbound	0321cfcbecb7f6e062f1d294	9	Service ID CT1	LINESTRING (103.76499 1.46354, 103.76472 1.46404, 103.
4	Route_03418837d7e991a5eebb9f42_inbound	03418837d7e991a5eebb9f42	7	Service ID P213	LINESTRING (103.62827 1.53824, 103.6282 1.5383, 103.62
5	Route_03418837d7e991a5eebb9f42_outbound	03418837d7e991a5eebb9f42	7	Service ID P213	LINESTRING (103.5898 1.44678, 103.58984 1.44703, 103.5
6	Route_08862d50c23665b56e6e1fbf_inbound	08862d50c23665b56e6e1fbf	25	Service ID 331	LINESTRING (103.62827 1.53824, 103.6282 1.5383, 103.62
7	Route_08862d50c23665b56e6e1fbf_outbound	08862d50c23665b56e6e1fbf	25	Service ID 331	LINESTRING (103.76499 1.46354, 103.76472 1.46404, 103.

- Uploaded via Moata API to our SQL database in the cloud
- Layers are configured to point towards the relevant assets in the database

Application

Third party APIs

- Query API in Python and process to tabular format with text-represented geometries

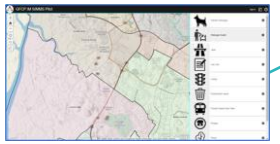

```
{
   status: "ok",
   data: {
      aqi: 70,
      time: {
         s: "2021-06-20 09:00:00"
      },
      city: {
         name: "Shanghai",
         url: "http://aqicn.org/city/shanghai/",
         geo: [
            "31.2047372",
            "121.4489017"
         ]
      },
      iaqi: {
         pm25: "..."
      }
   }
}
```


	A	B	C	D	E
1	Name	Today's AQI	WKT		
2	Shanghai	70	POINT(121.4489017 31.2047372)		

- As before, upload to database and configure layers to point to the relevant assets
- Use of Microsoft Azure services to host the Python code on the cloud and run once per day

Data collection template preparation from the pilot project

- Citizen Feedback Portal
- Traffic Light Survey
- Traffic Count Portal



3. Apps Interface

4. User submit data

Synchronize Changes

Map

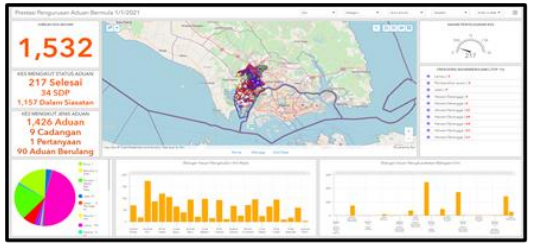
2. Database

Publish

View, Filter, Edit

1.Prepare Data Template

5.Dashboard



FUTURE CITIES
Global Future Cities Programme

Practical

Practical

Template for Field worker to verify road in study area

Use QGIS & Online GIS to run some examples

Preparation template for data collection

- Add data
- Add fields
- Field type and properties
- Field names
- Intersect
- Publish data as a hosted feature service (URL link provided for SIMMS)
- Collector/Web Application for field worker

Tutorial Data: https://arcgis.com/home/item.html?id=220c18e8208d40deb21a97d609b6e5d9

Summary

Summary

Theory

- GIS Fundamentals (best practice, naming conventions, data formats)
- Online storage and interaction (direct links, APIs)

Application (via SIMMS)

- ArcGIS servers (Moata Platform)
- Data collection

Practical

- Basic GIS operations
- Publishing to ArcGIS Online for data collection

FUTURE CITIES

Global Future Cities Programme

Thank you