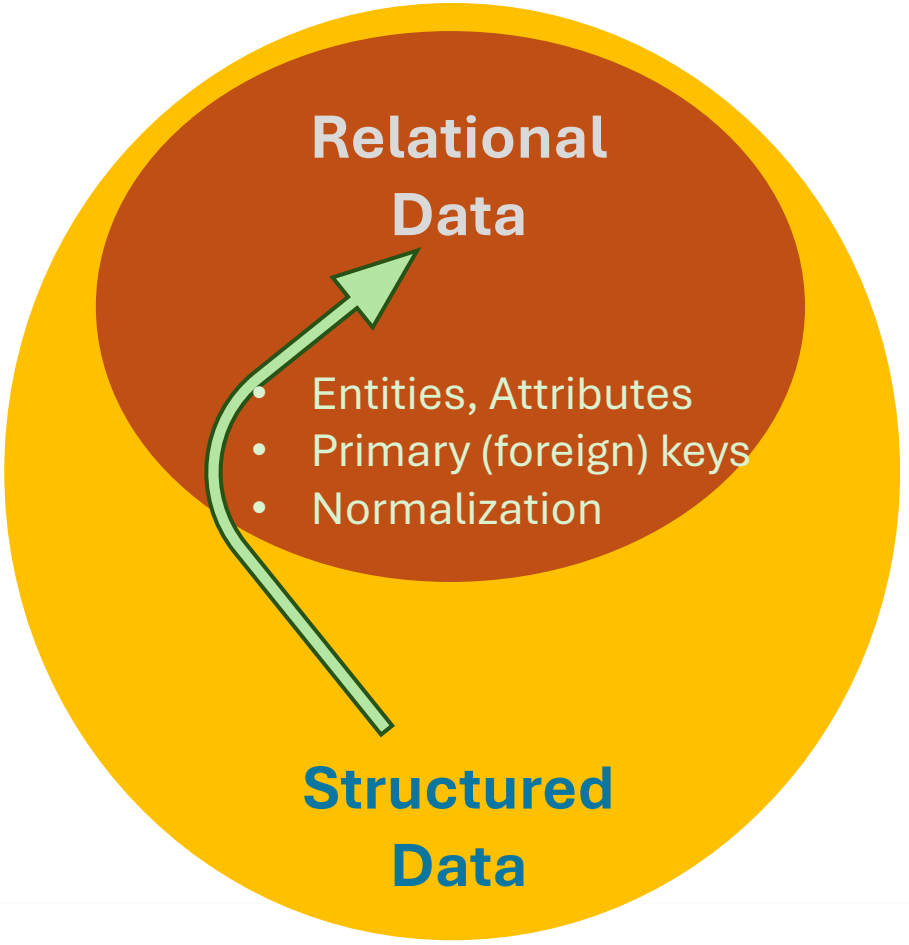
Semi-structured Data & JSON

1405 Instructor: Ruiqing (Sam) Cao

Structured Data & Relational Database

- Structured data: Information organized in tabular format (e.g., spreadsheet), i.e., in rows and columns
- Relational data: data represented as *relations*, where a relation is a set of tuples sharing the same attributes

  → Structured data organized into tables, where relationships are explicitly defined using keys



Dealing With Semi- & Unstructured Data

- Review: many forms of data cannot be entirely structured within a pre-defined data model
- Unstructured data: e.g., text, image, audio, video…
- Semi-structured data: have some level of organization (e.g., HTML/XML tags or markers) but does not strictly conform to a rigid schema like structured data

Common Types of Semi-Structured Data

HTML/XML

```
<html xmlns="http://www.w3.org/1999/xhtml" ng-app="CourseDescriptionApp" ng-
controller="CourseDescriptionCtrl" ng-cloak>
<head>
    <title>
        Stockholm School of Economics
    </title>

    <meta name="viewport" content="initial-scale=1.0, user-scalable=0, minimum-scale=1.0, maximum-
scale=1.0, width=device-width" />

    <link rel="icon" href="/css/img/favicon.ico" type="image/vnd.microsoft.icon" />
    <link rel="apple-touch-icon-precomposed" href="/css/img/apple-touch-icon.png" />
    <link rel="stylesheet" type="text/css" href="/css/bundle.css" />
    <link rel="stylesheet" href="https://use.fontawesome.com/e0806b813c.css">
    <script src="/js/angular.min.js?rev=1" type="text/javascript"></script>
    <script src="/js/Show.js?rev=1" type="text/javascript"></script>


    <meta charset="utf-8" />

    <meta name="description" content="Description of course/module {{courseDescrip
the Stockholm School of Economics" />
    <meta property="og:title" content="{{courseDescription.PublicSubHeader}}" />
    <meta property="og:description" content="Description of course/module
{{courseDescription.CourseNr}} at the Stockholm School of Economics" />
</head>
```

Tags and markers indicate structure

JSON


```
[{"@context":"http://schema.org","@type":"WebApplication","name":"Notion
World","description":"Discover the powerful world of Notion with this free
directory of the best resources and tools about Notion. Whether you are a
beginner or want to boost your Notion skills, this curated list will help
you find everything you need to make you a Notion master.","datePublished":
"2022-09-26T13:34:12.815-07:00","aggregateRating":{"@type":"AggregateRating
","ratingCount":6,"ratingValue":"5.0","worstRating":1,"bestRating":5},"offe
rs":{"@type":"Offer","price":0,"priceCurrency":"USD"},"applicationCategory"
:"Productivity"},{"@context":"http://schema.org","@type":"BreadcrumbList","
itemListElement":[{"@type":"ListItem","position":1,"name":"Home","item":"ht
tps://www.producthunt.com/"},{"@type":"ListItem","position":2,"name":"Notio
n World","item":"https://www.producthunt.com/products/notion-world"}]}]
```


Nested lists and dictionaries

Common Types of Semi-Structured Data

- HTML and XML are date objects that have some structure but are more flexible than actually structured data
- JSON (Java Script Object Notation) is a very popular format used in many different Internet companies

➔Their structures can be represented by trees

➔Access to data is navigational through the root-to-node path to retrieve information

Exercise: Tree Structure of JSON Data

Product information in JSON format


```
{"product": {"__typename": "Product", "id": "106091", "slug": "notion", "reviewsCount": 2805, "addonsCount": 35, "canClaim":
false, "badges": {"__typename": "Connection", "totalCount": 38}, "shoutoutsToCount": 797, "name": "Notion", "tagline": "The
all-in-one workspace", "isNoLongerOnline": false, "canEdit": false, "followersCount": 10956, "activeUpcomingEvent": null,
"upcomingBannerFollowers": {"__typename": "UserConnection", "edges": [{"__typename": "UserEdge", "node": {"__typename":
"User", "id": "105600", "name": "Farooq (SF Ali) Zafar", "username": "sfali789", "avatarUrl": "https://ph-avatars.imgix.net/
105600/1b2f2e5c-2281-4acb-9777-3870f97e90e8.jpeg"}}, {"__typename": "UserEdge", "node": {"__typename": "User", "id": "18280",
"name": "Chris Messina", "username": "chrismessina", "avatarUrl": "https://ph-avatars.imgix.net/18280/f49f2882-55cd-4279-
add0-7761638c8104.png"}}, {"__typename": "UserEdge", "node": {"__typename": "User", "id": "2", "name": "Ryan Hoover",
"username": "rrhoover", "avatarUrl": "https://ph-avatars.imgix.net/2/original.jpeg"}}]}, "alternativesCount": 304,
"targetedAd": null, "followers": {"__typename": "UserConnection", "edges": [{"__typename": "UserEdge", "node": {"__typename":
"User", "id": "6775383", "name": "Amartya Jha", "username": "amartya_jha", "avatarUrl": "https://ph-avatars.imgix.net/
6775383/c8507d7a-b7d4-4504-bf88-178a37c314a9.png"}}, {"__typename": "UserEdge", "node": {"__typename": "User", "id":
"7624751", "name": "Anbang Xu", "username": "anbang_xu", "avatarUrl": "https://ph-avatars.imgix.net/7624751/6f57ce23-
a692-4a00-828a-39b8278011d3.webp"}}, {"__typename": "UserEdge", "node": {"__typename": "User", "id": "7684967", "name":
"Nikhilesh", "username": "nikhilesh1", "avatarUrl": "https://ph-avatars.imgix.net/7684967/original.png"}}, {"__typename":
"UserEdge", "node": {"__typename": "User", "id": "7682245", "name": "Ron Kolk", "username": "ron_kolk", "avatarUrl":
"https://ph-avatars.imgix.net/7682245/original.jpeg"}}, {"__typename": "UserEdge", "node": {"__typename": "User", "id":
"7683318", "name": "Paislee Burke", "username": "paislee_burke", "avatarUrl": "https://ph-avatars.imgix.net/7683318/
original.png"}}, {"__typename": "UserEdge", "node": {"__typename": "User", "id": "6102869", "name": "Healsha", "username":
"yiord", "avatarUrl": "https://ph-avatars.imgix.net/6102869/cd5ac877-1d0d-4705-87d0-33e23e9bb8ba.jpeg"}}, {"__typename":
"UserEdge", "node": {"__typename": "User", "id": "7468137", "name": "Ivan Zabudko", "username": "ivan_zabudko", "avatarUrl":
"https://ph-avatars.imgix.net/7468137/b65eeea6-75fa-4163-aa3b-279a70c4c6a5.png"}}, {"__typename": "UserEdge", "node":
{"__typename": "User", "id": "7681247", "name": "George Norris", "username": "georgenorris", "avatarUrl": "https://ph-
avatars.imgix.net/7681247/original.jpeg"}}, {"__typename": "UserEdge", "node": {"__typename": "User", "id": "4964359",
"name": "Ibrahem Alhofe", "username": "ibrahem_alhofe", "avatarUrl": "https://ph-avatars.imgix.net/4964359/original.png"}},
{"__typename": "UserEdge", "node": {"__typename": "User", "id": "7492589", "name": "Ashwin Bhat", "username": "ashwin_bhat3",
"avatarUrl": "https://ph-avatars.imgix.net/7492589/43637b4b-6347-4325-9b3d-e3206c04b9d7.jpeg"}}, {"__typename": "UserEdge",
"node": {"__typename": "User", "id": "7663698", "name": "martin teira", "username": "martin_teira", "avatarUrl": "https://ph-
avatars.imgix.net/7663698/original.jpeg"}}, {"__typename": "UserEdge", "node": {"__typename": "User", "id": "7677911",
"name": "Peter Josh", "username": "peter_josh", "avatarUrl": "https://ph-avatars.imgix.net/7677911/original.jpeg"}},
{"__typename": "UserEdge", "node": {"__typename": "User", "id": "7677595", "name": "Frank Thompson", "username":
"frank_thompson2", "avatarUrl": "https://ph-avatars.imgix.net/7677595/original.png"}}, {"__typename": "UserEdge", "node":
{"__typename": "User", "id": "4201903", "name": "Martin Gugel", "username": "martin_gugel", "avatarUrl": "https://ph-
avatars.imgix.net/4201903/f00fab2b-651c-4041-8ae1-6b1cbe9d7ead.jpeg"}}, {"__typename": "UserEdge", "node": {"__typename":
"User", "id": "7676981", "name": "ASV-R Official", "username": "asv_r_official", "avatarUrl": "https://ph-avatars.imgix.net/
7676981/original.png"}}, {"__typename": "UserEdge", "node": {"__typename": "User", "id": "6959271", "name": "Hannah Ryan",
```


In command line:

1. Install `pyjsonviewer` with `pip install pyjsonviewer`
2. Visualize the JSON file with `pyjsonviewer -f example.json`

Viewing JSON's Tree Structure

In command line:

1. Install `pyjsonviewer` with

```
pip install pyjsonviewer
```

2. Visualize the JSON file with

```
pyjsonviewer -f example.json
```

Viewing JSON's Tree Structure

In command line:

1. Install `pyjsonviewer` with

```
pip install pyjsonviewer
```

2. Visualize the JSON file with

```
pyjsonviewer -f example.json
```

The data consists of nested lists and dictionaries

Viewing JSON's Tree Structure

In Python:

1. Import the `json` module

```
import json
```

2. Load the JSON data object

```
with open('example.json','r') as f:
    x = json.loads(f.read())
```

The loaded JSON object is a nested dictionary stored in **x** which has the type `dict()`


```
{'product': {'__typename': 'Product',
  'id': '106091',
  'slug': 'notion',
  'reviewsCount': 2805,
  'addonsCount': 35,
  'canClaim': False,
  'badges': {'__typename': 'Connection', 'totalCount': 38},
  'shoutoutsToCount': 797,
  'name': 'Notion',
  'tagline': 'The all-in-one workspace',
  'isNoLongerOnline': False,
  'canEdit': False,
  'followersCount': 10956,
  'activeUpcomingEvent': None,
  'upcomingBannerFollowers': {'__typename': 'UserConnection',
   'edges': [{'__typename': 'UserEdge',
     'node': {'__typename': 'User',
      'id': '105600',
      'name': 'Farooq (SF Ali) Zafar',
      'username': 'sfali789',
      'avatarUrl': 'https://ph-avatars.imgix.net/105600/1b2f2e5c-2281-4acb-9777-3870f97e90e8.jpeg'}},
    {'__typename': 'UserEdge',
     'node': {'__typename': 'User',
      'id': '18280',
      'name': 'Chris Messina',
      'username': 'chrismessina',
      'avatarUrl': 'https://ph-avatars.imgix.net/18280/f49f2882-55cd-4279-add0-7761638c8104.png'}},
    {'__typename': 'UserEdge',
     'node': {'__typename': 'User',
      'id': '2',
      'name': 'Ryan Hoover',
      'username': 'rrhoover',
      'avatarUrl': 'https://ph-avatars.imgix.net/2/original.jpeg'}}]},
  'alternativesCount': 304,
  'targetedAd': None,
  'followers': {'__typename': 'UserConnection',
   'edges': [{'__typename': 'UserEdge',
     'node': {'__typename': 'User',
      'id': '6775383',
      'name': 'Amartya Jha',
      'username': 'amartya_jha',
      'avatarUrl': 'https://ph-avatars.imgix.net/6775383/c8507d7a-b7d4-4504-bf88-178a37c314a9.png'}},
    {'__typename': 'UserEdge',
```