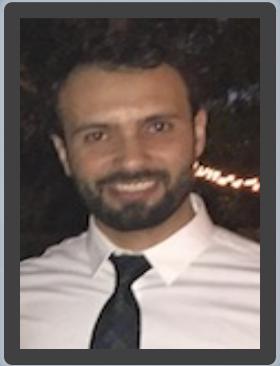




# UNIVERSAL BASIC INCOME ANALYSIS



PRESENTED BY HATEM AL-GHUTI

# NUMISMATICS & UNIVERSAL BASIC INCOME ANALYSIS

# **SUMMARY**

## Numismatic tools for Analyzing Universal Basic Income

---

Objective- develop tool for analyzing the feasibility of Universal Basic Income. The Financial problem which will be analyzed are presented by certain requirements related to Shariah based monetary systems— affecting approximately 24% of the world population.

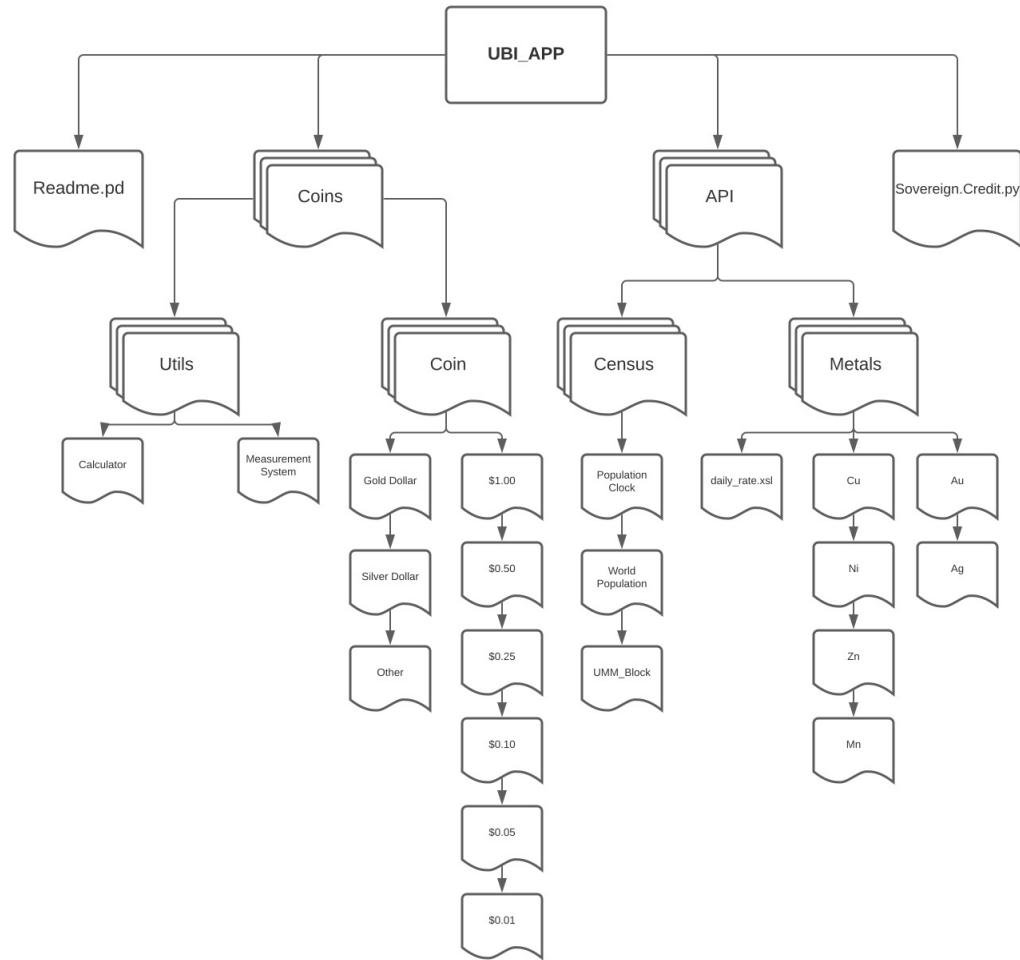
**PART 1- Overview & App Installation**

**PART 2- Database & API Integration**

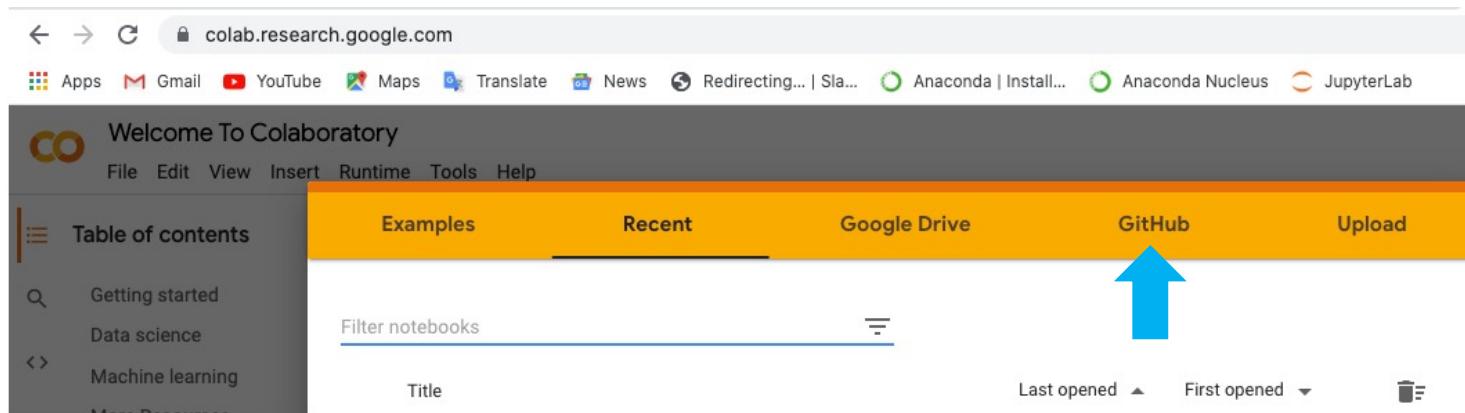
**PART 2- Market Overview**

# PART 1- Overview & Installation

# OVERVIEW



# Installation



# Calculator App

The screenshot shows a web browser window with the title bar "calculator" and the address bar "localhost:8866". The page content is titled "Seigneurage Calculator". It provides instructions for getting the current melt value for a coin in the U.S. catalogue list by entering the name of the coin (e.g., dollar, quarter, dime, nickel, penny) and running the command. A note states that the calculator is case sensitive and provides examples for multiple coins separated by a plus sign. Below the instructions, a "Current List" is provided:

- dollar**: 13.883422905062925
- dime**: 0.072028346816067
- quarter**: 0.0579454802012737
- penny**: 0.023178192080509474
- nickel**: 0.059581497019754015
- dollar + quarter + dime + nickel + penny**: 0.22167282397818444

# PART 1- Organizing Data

## Database & API Integration

# Coins

Denomination	Cent	Nickel	Dime	Quarter Dollar	Half Dollar	Dollar
Composition	Copper Plated Zinc 2.5% Cu Balance Zn	Cupro-Nickel 25% Ni Balance Cu	Cupro-Nickel 8.33% Ni Balance Cu	Cupro-Nickel 8.33% Ni Balance Cu	Cupro-Nickel 8.33% Ni Balance Cu	Manganese-Brass 88.5% Cu 6% Zn 3.5% Mn 2% Ni
Weight	2.500 g	5.000 g	2.268 g	5.670 g	11.340 g	8.1 g
Diameter	0.750 in. 19.05 mm	0.835 in. 21.21 mm	0.705 in. 17.91 mm	0.955 in. 24.26 mm	1.205 in. 30.61 mm	1.043 in. 26.49 mm
Thickness	1.52 mm	1.95 mm	1.35 mm	1.75 mm	2.15 mm	2.00 mm
Edge	Plain	Plain	Reeded	Reeded	Reeded	Edge-Lettering
No. of Reeds	N/A	N/A		118	119	150

## PART 1: MEASUREMENT VARIABLES

```
In [4]: avdp_ou = 28.34952312
```

```
#store avdp_ou
```

```
Stored 'avdp_ou' (float)
```

```
In [5]: troy_ou = 31.1034768
```

```
In [6]: metric_tonne = 1000000
```

```
In [7]: gram = 1
```

## PART 2: COIN LIST & WEIGHTS

```
In [8]: # provide the weight (in grams) for each of the coins listed in the US mint catalog.
```

```
Penny_wt = 2.5*gram
Nickel_wt = 5.0*gram
Dime_wt = 2.268*gram
Quarter_wt = 5.67*gram
Half_Dollar_wt = 11.34*gram
dollar_coin_wt = 8.1*gram
```

# API INTEGRATION

```
In [9]: # The Population Calculator & Metal Prices API Call endpoint URLs  
  
Population_url = "https://api.census.gov/data/2019/pep/population"  
  
Metals_url = "https://www.metals-api.com/api/latest?access_key=g7t3011a5xvpn5e3d22fg95dc5zzom6epc728v26y870gu4mp38o7rn0aiv5&base=USD&symbol=XAG,XAU,XCU,ZNC"
```

Step 2: Using the Python requests library, make an API call to access the current price USD for metals

```
In [10]: response = requests.get(Metals_url).json()
```

Step 3: Use the json.dumps function to review the response data from the API call

To-Do: Use the indent and sort\_keys parameters to make the response object readable

```
In [11]: print(json.dumps(response, indent=4, sort_keys=True))
```

```
{  
    "base": "USD",  
    "date": "2021-12-21",  
    "rates": {  
        "NI": 1.646374403514,  
        "USD": 1,  
        "XAG": 0.0449190456,  
        "XAU": 0.00055830132,  
        "XCU": 3.7126901669759,  
        "ZNC": 10.379441282259  
    },  
    "success": true,  
    "timestamp": 1640064780,  
    "unit": "per ounce"  
}
```

```
# Copper Price per Avdp Ounce  
  
cu_rates = response['rates']['XCU']  
USD = response['rates']['USD']  
  
cu_avdp_ou = USD/cu_rates  
  
print(f"Copper avdp ou ${cu_avdp_ou: .2f}")
```

```
# Nickel Price per Avdp Ounce  
ni_rates = response['rates']['NI']  
USD = response['rates']['USD']  
  
ni_troy_ou = USD/ni_rates  
  
print(f"Nickel avdp ou ${ni_troy_ou: .5f}")  
  
%store USD
```

```
zn_rates = response['rates']['ZNC']  
USD = response['rates']['USD']  
  
zn_avdp_ou = USD/zn_rates  
  
print(f"Zinc avdp ou ${zn_avdp_ou: .5f}")
```

Note: No central exchange has ever existed for setting the price of manganese ore. Rather, prices have been established by negotiation between buyers and sellers, taking into account such factors as manganese content, content of other elements, physical characteristics, quantity, and, of considerable significance, ocean freight rates. <https://pubs.usgs.gov/sir/2012/5188/sir2012-5188.pdf#Manganese>

## **PART 2-** The Opportunity

# OVERVIEW

## MUSLIM MAJORITY NATIONS

### Key Player- Secured Digital Currency

1. Kinesis

### Fifty-Seven Member of OIC

**Population-** 1.82 Billion  
**GDP -** \$27.95 Trillion (est. 2019)  
**PPP-** \$19,500 (est.)

## Targeted Data Analysis

### Islamic Cooperative Population

```
In [52]: df= International_Database.iloc[0:227]

In [53]: rows= [1,2,3,16, 17, 22, 26, 36, 40, 44, 56, 60, 72, 73, 85, 86, 94, 95, 96, 104, 105, 111, 112, 115, 118, 125, 126, 127, 130, 139, 140, 1

In [54]: umm_population= df["Population"]

In [55]: selected_population= umm_population.loc[rows].sum()

In [56]: print(selected_population)

1827880190

In [57]: ratio= selected_population / world_population

In [58]: print(f"{ratio: .2f}")

0.24
```



### TOP 10 COUNTRIES, BY GDP (PPP) IN 2030



# **THANK YOU**