

Bitcoin Sovellus

Tässä tehtävässä tulee toteuttaa sovellus, joka auttaa Roope Ankkaa analysoimaan bitcoinin markkinahintoja tiettyjen päivämääräväliin perustuen.

Sovelluksesta Lisätietoa

Sovelluksessa tulee olla:

1. Analysoitavat aikavälit: Käyttäjän tulee pystyä syöttämään päivämäärävälin.
2. Päivän hinta: Päivän hinta tarkoittaa hintaa klo 00:00 UCT. Jos tarkkaa datapistettä ei ole, käytä mahdollisimman lähellä keskiyötä olevaa tietoa.

Tehtävät

A. Päivät, joilla alin ja korkein hinta (T1)

- Tulos: Päivät, joilla bitcoinin markkinahinta oli alimmillaan ja korkeimmillaan annetulla aikavälillä. (Euroina)

B. Päivät, joilla alin ja korkein kaupankäyntivolyymi (T1)

- Tulos: Päivät, joilla kaupankäyntivolyymi oli alhaisin ja korkein annetulla aikavälillä. (Euroina)

C. Pisin laskeva ja nouseva hinta (T2, H3)

- Määritelmät:
 - Laskeva hinta: Päivän N hinta on matalampi kuin päivän N-1 hinta.
 - Nouseva hinta: Päivän N hinta on korkeampi kuin päivän N-1 hinta.
- Tulos: Pisimmät peräkkäiset päivät, jolloin bitcoinin hinta joko laski tai nousi annetulla aikavälillä.

D.Paras päivä ostaa ja myydä (H4, K5)

- Selitys: Sovelluksen tulee tunnistaa päivämäärät, jolloin bitcoin kannattaa ostaa halvalla ja myydä kalliilla maksimaalisen voiton saamiseksi. Myös päinvastainen skenaario tulee huomioida.
- Tulos: Kaksi päivää: ostopäivä ja myyntipäivä molemmille skenaarioille.

Ohjeet

1. Käytä CoinGecko API:a:

- API palauttaa dataa eri tarkkuuksilla aikavälin pituuden mukaan.
- Lisää aina 1 tunti to-parametriin varmistaaksesi loppupäivän datan saatavuus.

2. Valuuttana käytetään euroa (€).

Arviointi

- Selkeä ja luettava koodi.
- Käytön helppous: Sovelluksen mukana tulee olla ohjeet sen suorittamiseen (README-tiedosto)
- Yksinkertaisuus: Ulkoisten kirjastojen minimointi. Pyri käyttämään vain yhden kolmannen osapuolen kirjastoa tarvittaessa.
- Laajennettavuus: Sovellus tulee olla helposti päivitettävä.

Tehtävä 2.2 Palautus

a) from ja to parametrien merkitys

CoinGecko API:ssa from ja to parametrit määrittelevät aikavälin, jonka sisällä hintatietoja haetaan. Näiden arvojen tulee olla UNIX-ajanlaskun mukaisia aikaleimoja.

b) Aikavälin asettaminen kahteen edeltävään viikkoon
Nykyhetken ja edeltävän kahden viikon aikaväli voidaan laskea seuraavasti:

- a. Hae nykyhetken nykyhetken UNIX-aikaleima.
- b. Laske 14 päivän aikaleima ($14 \text{ päivää} * 24 \text{ tuntia} * 60 \text{ minuuttia} * 60 \text{ sekuntia}$) ja vähennä se nykyhetken aikaleimasta.

c) Aikojen käsittely C#:ssa

Aikaleimojen käsittely voidaan toteuttaa seuraavasti:

- a. Nykyhetken UNIX-aikaleima: Käytä
`DateTimeOffset.Now.ToUnixTimeSeconds()`.
- b. Kahden viikon aikaleima: Vähennä nykyhetkestä
14 päivää ja muuta se UNIX-ajaksi:

```
long now =  
DateTimeOffset.Now.ToUnixTimeSeconds();  
long twoWeeksAgo =  
DateTimeOffset.Now.AddDays(-  
14).ToUnixTimeSeconds();
```

c. Kutsu API:a:

URL:

```
string baseUrl =  
"https://api.coingecko.com/api/v3/coins/bitcoin/  
market_chart/range";  
string url =  
$" {baseUrl}?vs_currency=eur&from={twoWeeksA  
go}&to={now}";
```
