

```
message: `Processed ${missingDates.length} missing dates`,
});

} catch (err) {
  error( Unexpected error: ${err.message} );
  return res.json({ error: Unexpected error: ${err.message} }, 500);
}
}
```

Below is a detailed Persian guide for setting up the three Appwrite functions in Appwrite Cloud, including the main AI analysis function (provided in the previous response), the hourly Bitcoin data function, and the daily Bitcoin data function. The guide covers creating a project, setting up collections, configuring environment variables, deploying functions, setting up Telegram integration, and testing the setup. It ensures that a user with these three code files and this guide can fully implement and run the system. The guide is written in Persian to match the language of the application and includes all necessary details, including how to acquire environment variables and configure the Telegram bot.

راهنمای جامع راهاندازی توابع تحلیل بیت‌کوین در Appwrite Cloud

این راهنما به شما کمک می‌کند تا سه تابع Appwrite (تحلیل هوش مصنوعی، جمع‌آوری داده‌های ساعتی بیت‌کوین، و جمع‌آوری داده‌های روزانه بیت‌کوین) را در Appwrite Cloud راه‌اندازی کنید. این سیستم داده‌های بیت‌کوین را جمع‌آوری کرده، تحلیل‌های تکنیکال و بنیادی تولید می‌کند و نتایج را به صورت پیام متنی و سند Word (.docx) به یک کانال تلگرام ارسال می‌کند. این راهنما شامل تمام مراحل، از ایجاد پروژه تا تنظیم متغیرهای محیطی، ایجاد مجموعه‌ها (Collections)، و استقرار توابع است.

پیش‌نیازها

قبل از شروع، اطمینان حاصل کنید که موارد زیر را دارید:

- ثبت‌نام کنید در <https://appwrite.io> در حساب Appwrite Cloud.
- جهت جمع‌آوری داده‌های بیت‌کوین API برای دریافت کلید: <https://min-api.cryptocompare.com> حساب CryptoCompare.
- Gemini مدل API برای دریافت کلید: <https://makersuite.google.com> حساب Google Cloud.
- Gemini به عنوان جایگزین API برای دریافت کلید: <https://avalai.ir> حساب AvalAI.
- ربات تلگرام: برای ارسال تحلیل‌ها به یک کانال تلگرام.
- برای استقرار توابع از طریق خط فرمان (اختیاری، اما توصیه می‌شود): **Node.js** و **Appwrite CLI**.
- فایل‌های کد: سه فایل کد ارائه‌شده (تحلیل هوش مصنوعی، داده‌های ساعتی، داده‌های روزانه).

مراحل راه‌اندازی

1. ایجاد پروژه در Appwrite Cloud

- **Appwrite Cloud** ورود به
- بروید و با حساب کاربری خود وارد شوید <https://cloud.appwrite.io>
- کلیک کنید **Create Project** روی
- (**BitcoinAnalysis** مثلاً) نامی برای پروژه انتخاب کنید
- به صورت خودکار تولید می‌شود. این را یادداشت کنید **Project ID** یک
- **API** ایجاد کنید
- بروید **Settings > API Keys** در داشبورد پروژه، به
- کلیک کنید **Create API Key** روی



• وارد کنید `BitcoinAnalysisKey` نامی مانند

• را انتخاب کنید **Scopes**

• `databases.read`

• `databases.write`

• `collections.read`

• `collections.write`

• `documents.read`

• `documents.write`

• این کلید برای دسترسی توابع به پایگاه داده لازم است. (`APPWRITE_API_KEY`) تولیدشده را کپی کنید API کلید

• را در جایی امن ذخیره کنید، زیرا پس از بستن صفحه قابل مشاهده نیست API نکته امنیتی: کلید

2. ایجاد پایگاه داده و مجموعه ها

شما نیاز به یک پایگاه داده با سه مجموعه دارید:

- `ai_signals`: برای ذخیره تحلیل های هوش مصنوعی
- `hourly_data`: برای داده های ساعتی بیت کوین
- `daily_data`: برای داده های روزانه بیت کوین

2.1. ایجاد پایگاه داده

• بروید **Database** در داشبورد پروژه، به

• کلیک کنید **Create Database** روی

• وارد کنید `BitcoinDB` نامی مانند

• (`DATABASE_ID`) تولیدشده را یادداشت کنید **Database ID**

2.2. `ai_signals` ایجاد مجموعه

• کلیک کنید **Create Collection** روی `BitcoinDB` در پایگاه داده

• وارد کنید `ai_signals` نامی مانند

• (`AI_SIGNALS_COLLECTION_ID`) را یادداشت کنید **Collection ID**

• **Permissions:**

• اضافه کنید `Role` را به عنوان یک API را انتخاب کنید یا کلید `Any` گزینه `Read` و `Write` برای

• **Attributes (ویژگی ها):**

• ویژگی های زیر را با نوع و تنظیمات مشخص شده اضافه کنید

نام ویژگی	نوع داده	اندازه	پیش فرض	ضروری	توضیحات
timestamp	String	50	-	خیر	(ISO 8601) زمان تولید تحلیل
signal	String	10	-	خیر	(LONG/SHORT/HOLD) سیگنال
confidence	String	10	-	خیر	اطمینان (کم/متوسط/بالا)
price	Double	-	-	خیر	قیمت فعلی بیت کوین
leverage	String	10	-	خیر	اهرم پیشنهادی (مثل 1:5)
analysis	String	10000	-	خیر	(JSON) تحلیل کامل
ai_provider	String	20	-	خیر	AI ارائه دهنده (Gemini/AvalAI)
success	Boolean	-	false	خیر	وضعیت موفقیت تحلیل
gemini_request_count	Integer	-	0	خیر	تعداد درخواست های Gemini

• **Indexes:**

• ایجاد کنید تا جستجوهای مبتنی بر زمان سریع تر شوند `timestamp` روی `Key` یک شاخص از نوع

2.3. hourly_data ایجاد مجموعه

- کلیک کنید **Create Collection** روی ، **BitcoinDB** در پایگاه داده
- وارد کنید **hourly_data** نامی مانند
- **Collection ID** (**HOURLY_COLLECTION_ID**) را پادداشت کنید
- **Permissions:**
- تنظیم کنید **Read** و **Write** را برای API یا کلید **Any** گزینه ، **ai_signals** مشابه
- **Attributes:**
- ویژگی‌های زیر را اضافه کنید (بر اساس کد ساعتی)

نام ویژگی	نوع داده	اندازه	پیش‌فرض	ضروری	توضیحات
date	String	50	-	خیر	(ISO 8601) تاریخ کندل
open	Double	-	-	خیر	قیمت باز شدن
high	Double	-	-	خیر	بالا ترین قیمت
low	Double	-	-	خیر	پایین ترین قیمت
close	Double	-	-	خیر	قیمت بسته شدن
volume	Double	-	-	خیر	حجم معاملات (BTC)
pct_change_1h_ago	Double	-	-	خیر	تغییر درصد 1 ساعت قبل
pct_change_6h_ago	Double	-	-	خیر	تغییر درصد 6 ساعت قبل
pct_change_24h_ago	Double	-	-	خیر	تغییر درصد 24 ساعت قبل
pct_change_1h_future	Double	-	-	خیر	تغییر درصد 1 ساعت آینده
pct_change_6h_future	Double	-	-	خیر	تغییر درصد 6 ساعت آینده
pct_change_24h_future	Double	-	-	خیر	تغییر درصد 24 ساعت آینده
rsi	Double	-	-	خیر	(RSI) شاخص قدرت نسبی
ema_50h	Double	-	-	خیر	ساعته 50 EMA
sma_20h	Double	-	-	خیر	ساعته 20 SMA
sd_20h	Double	-	-	خیر	انحراف معیار 20 ساعته
upper_band_20h	Double	-	-	خیر	باند بولینگر بالا
lower_band_20h	Double	-	-	خیر	باند بولینگر پایین
ema_26h	Double	-	-	خیر	ساعته 26 EMA
ema_12h	Double	-	-	خیر	ساعته 12 EMA
macd_line	Double	-	-	خیر	خط MACD
macd_signal_line	Double	-	-	خیر	خط سیگنال MACD
macd_histogram	Double	-	-	خیر	هیستوگرام MACD
high_24h	Double	-	-	خیر	بالا ترین قیمت 24 ساعته
low_24h	Double	-	-	خیر	پایین ترین قیمت 24 ساعته
fib_24h_0_23	Double	-	-	خیر	فیوناچی 0.23 (24 ساعته)
fib_24h_0_38	Double	-	-	خیر	فیوناچی 0.38 (24 ساعته)



فیوناچی 0.5 (24 ساعته)	خیر	-	-	Double	fib_24h_0_5
فیوناچی 0.61 (24 ساعته)	خیر	-	-	Double	fib_24h_0_61
فیوناچی 0.78 (24 ساعته)	خیر	-	-	Double	fib_24h_0_78
بالاترین قیمت 48 ساعته	خیر	-	-	Double	high_48h
پایین‌ترین قیمت 48 ساعته	خیر	-	-	Double	low_48h
فیوناچی 0.23 (48 ساعته)	خیر	-	-	Double	fib_48h_0_23
فیوناچی 0.38 (48 ساعته)	خیر	-	-	Double	fib_48h_0_38
فیوناچی 0.5 (48 ساعته)	خیر	-	-	Double	fib_48h_0_5
فیوناچی 0.61 (48 ساعته)	خیر	-	-	Double	fib_48h_0_61
فیوناچی 0.78 (48 ساعته)	خیر	-	-	Double	fib_48h_0_78
بالاترین قیمت 72 ساعته	خیر	-	-	Double	high_72h
پایین‌ترین قیمت 72 ساعته	خیر	-	-	Double	low_72h
فیوناچی 0.23 (72 ساعته)	خیر	-	-	Double	fib_72h_0_23
فیوناچی 0.38 (72 ساعته)	خیر	-	-	Double	fib_72h_0_38
فیوناچی 0.5 (72 ساعته)	خیر	-	-	Double	fib_72h_0_5
فیوناچی 0.61 (72 ساعته)	خیر	-	-	Double	fib_72h_0_61
فیوناچی 0.78 (72 ساعته)	خیر	-	-	Double	fib_72h_0_78
هفته‌ای SMA 20	خیر	-	-	Double	sma_20w
هفته‌ای EMA 21	خیر	-	-	Double	ema_21w
وجود شکاف ارزش منصفانه صعودی	خیر	false	-	Boolean	fvg_bullish_exists
محدوده شکاف صعودی	خیر	-	50	String	fvg_bullish_range
وجود شکاف ارزش منصفانه نزولی	خیر	false	-	Boolean	fvg_bearish_exists
محدوده شکاف نزولی	خیر	-	50	String	fvg_bearish_range
OBV (حجم تعادلی)	خیر	-	-	Double	obv
میانگین قیمت وزنی حجمی	خیر	-	-	Double	vwap
محدوده واقعی	خیر	-	-	Double	true_range
میانگین محدوده واقعی	خیر	-	-	Double	atr
K استوکاستیک	خیر	-	-	Double	stochastic_k
D استوکاستیک	خیر	-	-	Double	stochastic_d

Indexes: •

• ایجاد کنید **date** روی **Key** یک شاخص از نوع

2.4. ایجاد مجموعه **daily_data**

• کلیک کنید **Create Collection** روی **BitcoinDB** در پایگاه داده

• وارد کنید **daily_data** نامی مانند

• (**DAILY_COLLECTION_ID**) را یادداشت کنید **Collection ID**

Permissions: •

• .را تنظیم کنید API یا کلید Any گزینه ، hourly_data و ai_signals مشابه

Attributes: •

• ویژگی‌های زیر را اضافه کنید (بر اساس کد روزانه)

نام ویژگی	نوع داده	اندازه	پیش‌فرض	ضروری	توضیحات
date	String	50	-	خیر	تاریخ کنل (ISO 8601)
open	Double	-	-	خیر	قیمت باز شدن
high	Double	-	-	خیر	بالا ترین قیمت
low	Double	-	-	خیر	پایین ترین قیمت
close	Double	-	-	خیر	قیمت بسته شدن
volume	Double	-	-	خیر	حجم معاملات (BTC)
percentage_change_from_1_day_ago	Double	-	-	خیر	تغییر درصد 1 روز قبل
percentage_change_from_7_days_ago	Double	-	-	خیر	تغییر درصد 7 روز قبل
percentage_change_from_30_days_ago	Double	-	-	خیر	تغییر درصد 30 روز قبل
percentage_change_1_day_from_now	Double	-	-	خیر	تغییر درصد 1 روز آینده
percentage_change_7_days_from_now	Double	-	-	خیر	تغییر درصد 7 روز آینده
percentage_change_30_days_from_now	Double	-	-	خیر	تغییر درصد 30 روز آینده
rsi	Double	-	-	خیر	شاخص قدرت نسبی (RSI)
50_ema	Double	-	-	خیر	روژه 50 EMA
20_sma_middle_band	Double	-	-	خیر	روژه 20 SMA
20_period_sd	Double	-	-	خیر	انحراف معیار 20 روزه
upper_band	Double	-	-	خیر	باند بولینگر بالا
lower_band	Double	-	-	خیر	باند بولینگر پایین
26_ema	Double	-	-	خیر	روژه 26 EMA
12_ema	Double	-	-	خیر	روژه 12 EMA
macd_line	Double	-	-	خیر	خط MACD
macd_signal_line	Double	-	-	خیر	خط سیگنال MACD
macd_histogram	Double	-	-	خیر	هیستوگرام MACD
last_20_day_high	Double	-	-	خیر	بالا ترین قیمت 20 روزه



پایین‌ترین قیمت 20 روزه	خیر	-	-	Double	last_20_day_low
فیوناچی 0.23 (20 روزه)	خیر	-	-	Double	20_day_fib_23
فیوناچی 0.38 (20 روزه)	خیر	-	-	Double	20_day_fib_38
فیوناچی 0.5 (20 روزه)	خیر	-	-	Double	20_day_fib_50
فیوناچی 0.61 (20 روزه)	خیر	-	-	Double	20_day_fib_61
فیوناچی 0.78 (20 روزه)	خیر	-	-	Double	20_day_fib_78
بالا‌ترین قیمت 40 روزه	خیر	-	-	Double	last_40_day_high
پایین‌ترین قیمت 40 روزه	خیر	-	-	Double	last_40_day_low
فیوناچی 0.23 (40 روزه)	خیر	-	-	Double	40_day_fib_23
فیوناچی 0.38 (40 روزه)	خیر	-	-	Double	40_day_fib_38
فیوناچی 0.5 (40 روزه)	خیر	-	-	Double	40_day_fib_50
فیوناچی 0.61 (40 روزه)	خیر	-	-	Double	40_day_fib_61
فیوناچی 0.78 (40 روزه)	خیر	-	-	Double	40_day_fib_78
بالا‌ترین قیمت 60 روزه	خیر	-	-	Double	last_60_day_high
پایین‌ترین قیمت 60 روزه	خیر	-	-	Double	last_60_day_low
فیوناچی 0.23 (60 روزه)	خیر	-	-	Double	60_day_fib_23
فیوناچی 0.38 (60 روزه)	خیر	-	-	Double	60_day_fib_38
فیوناچی 0.5 (60 روزه)	خیر	-	-	Double	60_day_fib_50
فیوناچی 0.61 (60 روزه)	خیر	-	-	Double	60_day_fib_61
فیوناچی 0.78 (60 روزه)	خیر	-	-	Double	60_day_fib_78
هفته‌ای SMA 20	خیر	-	-	Double	20_week_sma_bullmarketsupportband
هفته‌ای EMA 21	خیر	-	-	Double	21_week_ema_bullmarketsupportband



bullish_fvg_exists	Boolean	-	false	خیر	وجود شکاف ارزش منصفانه صعودی
bullish_fvg_range	String	50	-	خیر	محدوده شکاف صعودی
bearish_fvg_exists	Boolean	-	false	خیر	وجود شکاف ارزش منصفانه نزولی
bearish_fvg_range	String	50	-	خیر	محدوده شکاف نزولی
obv	Double	-	-	خیر	حجم تعادلی (OBV)
vwap	Double	-	-	خیر	میانگین قیمت وزنی حجمی
true_range	Double	-	-	خیر	محدوده واقعی
atr	Double	-	-	خیر	میانگین محدوده واقعی
k_percent	Double	-	-	خیر	K استوکاستیک
d_percent	Double	-	-	خیر	D استوکاستیک

Indexes:

- ایجاد کنید **date** روی **Key** یک شاخص از نوع

3. تنظیم ربات تلگرام

- ایجاد ربات تلگرام

- در تلگرام، به **@BotFather** پیام دهید
- را ارسال کنید **/start** دستور
- :را وارد کنید و مراحل را دنبال کنید **/newbot** دستور
- .انتخاب کنید **AidinAIBot** نامی مانند
- .نام کاربری مانند **@AidinAIBot** انتخاب کنید
- **Bot Token** (**TELEGRAM_BOT_TOKEN**) را کپی کنید
- ایجاد کانال تلگرام
- .یک کانال عمومی یا خصوصی ایجاد کنید (مثلاً **@AidinAIChannel**)
- :ربات خود را به عنوان مدیر به کانال اضافه کنید
- .ربات خود را انتخاب کنید **> Add Administrator > Administrators** > به تنظیمات کانال بروید
- .مجوزهای ارسال پیام و مدیریت پیام‌ها را فعال کنید
- :را دریافت کنید **Channel ID**
- .یک پیام به کانال ارسال کنید
- .پیام را به **@userinfobot** فوروارد کنید
- **TELEGRAM_CHANNEL_ID**) آیدی عددی کانال (مثلاً **-1001234567890**) را کپی کنید

4. دریافت متغیرهای محیطی

هر تابع نیاز به متغیرهای محیطی خاصی دارد. در زیر متغیرهای موردنیاز برای هر تابع و نحوه دریافت آنها آورده شده است:

4.1. (Aidin AI) تابع تحلیل هوش مصنوعی

- متغیرهای محیطی

متغیر	توضیحات	نحوه دریافت
-------	---------	-------------



در بخش تنظیمات، Appwrite از داشبورد پروژه	معمولاً Appwrite آدرس سرور <code>https://cloud.appwrite.io/v1</code>)	APPWRITE_ENDPOINT
از داشبورد پروژه، در بخش Settings Project ID	Appwrite شناسه پروژه	APPWRITE_PROJECT_ID
از بخش Settings > API Keys در داشبورد پروژه	برای دسترسی به پایگاه داده API کلید	APPWRITE_API_KEY
از <code>https://makersuite.google.com</code> ، در بخش API Keys	Google Gemini برای API کلید	GEMINI_API_KEY
از <code>https://avalai.ir</code> پس از ثبت نام و ایجاد API کلید	(Gemini جایگزین) AvalAI برای API کلید	AVALAI_API_KEY
از بخش Database در داشبورد، هنگام ایجاد پایگاه داده	(شناسه پایگاه داده BitcoinDB)	DATABASE_ID
از بخش Database > Collections ، هنگام ایجاد مجموعه <code>daily_data</code>	شناسه مجموعه داده های روزانه <code>(daily_data)</code>	DAILY_COLLECTION_ID
از بخش Database > Collections ، هنگام ایجاد مجموعه <code>hourly_data</code>	شناسه مجموعه داده های ساعتی <code>(hourly_data)</code>	HOURLY_COLLECTION_ID
از <code>@BotFather</code> در تلگرام	توکن ربات تلگرام	TELEGRAM_BOT_TOKEN
از <code>@userinfobot</code> پس از فوروارد کردن پیام از کانال	شناسه کانال تلگرام	TELEGRAM_CHANNEL_ID

- تولید کنید. API را فعال کنید و کلید API Gemini، بروید، یک پروژه ایجاد کنید Google Cloud Console به `GEMINI_API_KEY` نکته: برای ثبت نام کنید و کلید را دریافت کنید AvalAI در پلتفرم `AVALAI_API_KEY` برای

4.2. تابع داده های ساعتی بیت کوین

- متغیر های محیطی:

متغیر	توضیحات	نحوه دریافت
APPWRITE_ENDPOINT	معمولاً Appwrite آدرس سرور <code>https://cloud.appwrite.io/v1</code>)	در بخش تنظیمات، Appwrite از داشبورد پروژه
APPWRITE_PROJECT_ID	Appwrite شناسه پروژه	از داشبورد پروژه، در بخش Settings Project ID
APPWRITE_API_KEY	برای دسترسی به پایگاه داده API کلید	از بخش Settings > API Keys در داشبورد پروژه
DATABASE_ID	(شناسه پایگاه داده BitcoinDB)	از بخش Database در داشبورد، هنگام ایجاد پایگاه داده
HOURLY_COLLECTION_ID	شناسه مجموعه داده های ساعتی <code>(hourly_data)</code>	از بخش Database > Collections هنگام ایجاد مجموعه <code>hourly_data</code>
CRYPTOCOMPARE_API_KEY	CryptoCompare برای API کلید	از <code>https://min-api.cryptocompare.com</code> پس از API ثبت نام و ایجاد کلید

4.3. تابع داده های روزانه بیت کوین

- متغیر های محیطی:

متغیر	توضیحات	نحوه دریافت
-------	---------	-------------



در بخش تنظیمات، Appwrite از داشبورد پروژه	معمولاً Appwrite آدرس سرور (<code>https://cloud.appwrite.io/v1</code>)	APPWRITE_ENDPOINT
از داشبورد پروژه، در بخش Settings Project ID	Appwrite شناسه پروژه	APPWRITE_PROJECT_ID
از بخش Settings > API Keys در داشبورد پروژه	برای دسترسی به پایگاه داده API کلید	APPWRITE_API_KEY
از بخش Database در داشبورد، هنگام ایجاد پایگاه داده	(شناسه پایگاه داده BitcoinDB)	DATABASE_ID
از بخش Database Collections هنگام ایجاد مجموعه ، daily_data	(شناسه مجموعه داده‌های روزانه daily_data)	DAILY_COLLECTION_ID
از <code>https://min-api.cryptocompare.com</code> پس از API ثبت‌نام و ایجاد کلید	CryptoCompare برای API کلید	CRYPTOCOMPARE_API_KEY

- را API ثبت‌نام کنید، به داشبورد بروید و کلید `https://min-api.cryptocompare.com` در ، `CRYPTOCOMPARE_API_KEY` نکته: برای لازم است CryptoCompare ایجاد کنید. این کلید برای دریافت داده‌های کندل‌های ساعتی و روزانه از

5. Appwrite Cloud استقرار توابع در

هر تابع را به‌صورت جداگانه در Appwrite Cloud مستقر کنید. می‌توانید از داشبورد Appwrite یا Appwrite CLI استفاده کنید. در زیر مراحل برای هر دو روش توضیح داده شده است.

Appwrite استقرار از طریق داشبورد 5.1

تابع تحلیل هوش مصنوعی (Aidin AI)

- ایجاد تابع
 - بروید **Functions** در داشبورد پروژه، به
 - کلیک کنید **Create Function** روی
 - وارد کنید **AidinAI** نامی مانند
 - تنظیم کنید **Node.js 18** را روی **Runtime**
 - را یادداشت کنید **Function ID**
- آپلود کد
 - ذخیره کنید `index.js` فایل کد تحلیل هوش مصنوعی را در یک فایل به نام
 - ایجاد کنید `package.json` یک فایل `json`

```
{
  "name": "aidin-ai",
  "version": "1.0.0",
  "main": "index.js",
  "dependencies": {
    "node-appwrite": "^12.0.1",
    "@google/generative-ai": "^0.21.0",
    "axios": "^1.7.7",
    "docx": "^8.5.0",
    "form-data": "^4.0.0"
  }
}
```
- را آپلود کنید `index.js` تابع، کد **Code** در بخش
- اضافه کنید **Settings > Configuration > Package.json** را در بخش `package.json` فایل
- تنظیم متغیرهای محیطی
- متغیرهای زیر را اضافه کنید، **Settings > Variables** در بخش



- APPWRITE_ENDPOINT : `https://cloud.appwrite.io/v1`
- شناسه پروژه شما : `APPWRITE_PROJECT_ID`
- کلید API ایجادشده : `APPWRITE_API_KEY`
- کلید API Gemini : `GEMINI_API_KEY`
- کلید API AvalAI : `AVALAI_API_KEY`
- شناسه پایگاه داده : `DATABASE_ID` `BitcoinDB`
- شناسه مجموعه : `DAILY_COLLECTION_ID` `daily_data`
- شناسه مجموعه : `HOURLY_COLLECTION_ID` `hourly_data`
- توکن ربات تلگرام : `TELEGRAM_BOT_TOKEN`
- شناسه کانال تلگرام : `TELEGRAM_CHANNEL_ID`
- (Cron): تنظیم برنامه زمان‌بندی
- عبارت `* * * 6/* 0` را وارد کنید، **Schedule** > **Settings** در بخش
- می‌شود (UTC+3 یا فرض EEST، 03:00، 09:00، 15:00، 21:00 به وقت) این تنظیم باعث اجرای تابع هر 6 ساعت
- :استقرار تابع
- کلیک کنید **Deploy** روی
- به‌طور خودکار وابستگی‌ها را نصب کرده و تابع را مستقر می‌کند Appwrite

تابع داده‌های ساعتی بیت‌کوین

- ایجاد تابع
- کلیک کنید **Create Function** بروید و روی **Functions** به
- ،وارد کنید `HourlyBitcoinData` نامی مانند
- ،تنظیم کنید `Node.js 18` را روی **Runtime**
- ،را یادداشت کنید **Function ID**
- :آپلود کد
- ،ذخیره کنید `index.js` فایل کد ساعتی را در
- :ایجاد کنید `package.json` یک فایل
- `json`
- ،را مشابه بالا آپلود کنید `package.json` کد و
- :تنظیم متغیرهای محیطی
- :متغیرهای زیر را اضافه کنید

- APPWRITE_ENDPOINT : `https://cloud.appwrite.io/v1`
- شناسه پروژه شما : `APPWRITE_PROJECT_ID`
- کلید API : `APPWRITE_API_KEY`
- شناسه پایگاه داده : `DATABASE_ID` `BitcoinDB`
- شناسه مجموعه : `HOURLY_COLLECTION_ID` `hourly_data`
- کلید API CryptoCompare : `CRYPTOCOMPARE_API_KEY`
- (Cron): تنظیم برنامه زمان‌بندی
- عبارت `* * * * 0` را وارد کنید (اجرای هر ساعت، در دقیقه صفر)
- :استقرار تابع

```
{
  "name": "hourly-bitcoin-data",
  "version": "1.0.0",
  "main": "index.js",
  "dependencies": {
    "node-appwrite": "^12.0.1",
    "axios": "^1.7.7",
    "mathjs": "^12.0.0"
  }
}
```



- کلیک کنید **Deploy** روی

تابع داده‌های روزانه بیت‌کوین

- ایجاد تابع

- کلیک کنید **Create Function** بروید و روی **Functions** به

- وارد کنید **DailyBitcoinData** نامی مانند

- تنظیم کنید **18 Node.js** را روی **Runtime**

- را یادداشت کنید **Function ID**

- آپلود کد

- ذخیره کنید **index.js** فایل کد روزانه را در

- فایل **package.json**:

json

```
{
  "name": "daily-bitcoin-data",
  "version": "1.0.0",
  "main": "index.js",
  "dependencies": {
    "node-appwrite": "^12.0.1",
    "axios": "^1.7.7",
    "mathjs": "^12.0.0"
  }
}
```

- را آپلود کنید **package.json** کد و

- تنظیم متغیرهای محیطی

- متغیرهای زیر را اضافه کنید

- **APPWRITE_ENDPOINT**: <https://cloud.appwrite.io/v1>

- شناسه پروژه شما: **APPWRITE_PROJECT_ID**

- کلید API: **APPWRITE_API_KEY**

- شناسه پایگاه داده: **DATABASE_ID** BitcoinDB

- شناسه مجموعه: **DAILY_COLLECTION_ID** daily_data

- کلید API CryptoCompare: **CRYPTOCOMPARE_API_KEY**

- **(Cron)** تنظیم برنامه زمان‌بندی

- (UTC) اجرای روزانه در ساعت 00:00 عبارت **0 0 * * * 0** را وارد کنید

- استقرار تابع

- کلیک کنید **Deploy** روی

5.2. (اختیاری) Appwrite CLI استقرار با

- نصب **Appwrite CLI**:

bash

```
npm install -g @appwrite.io/cli
appwrite login
```

- وارد شوید Appwrite Cloud با استفاده از ایمیل و رمز عبور

- ایجاد ساختار پروژه

- (**daily-bitcoin-data** ، **hourly-bitcoin-data** ، **aidin-ai** مثلاً) برای هر تابع، یک پوشه جداگانه ایجاد کنید

- در هر پوشه

- را قرار دهید (کد تابع) **index.js** فایل

- را با وابستگی‌های مربوطه اضافه کنید **package.json** فایل

- استقرار توابع



- برای تابع تحلیل هوش مصنوعی:

bash



```
cd aidin-ai
appwrite init function
appwrite deploy function --functionId=<AIDIN_AI_FUNCTION_ID>
appwrite functions update --functionId=<AIDIN_AI_FUNCTION_ID> --schedule="0 */6 * * *"
```

- برای تابع داده‌های ساعتی:

bash



```
cd hourly-bitcoin-data
appwrite init function
appwrite deploy function --functionId=<HOURLY_FUNCTION_ID>
appwrite functions update --functionId=<HOURLY_FUNCTION_ID> --schedule="0 * * * *"
```

- برای تابع داده‌های روزانه:

bash



```
cd daily-bitcoin-data
appwrite init function
appwrite deploy function --functionId=<DAILY_FUNCTION_ID>
appwrite functions update --functionId=<DAILY_FUNCTION_ID> --schedule="0 0 * * *"
```

- تنظیم متغیرهای محیطی:

- متغیرهای محیطی را برای هر تابع اضافه کنید (مانند بالا). Appwrite در داشبورد

- CLI یا از

bash



```
appwrite functions updateVar --functionId=<FUNCTION_ID> --key=<KEY> --value=<VALUE>
```

6. تست سیستم

- بررسی داده‌های ساعتی و روزانه

- را با داده‌های کندل و شاخص‌های تکنیکال پر کند **hourly_data** تابع ساعتی باید هر ساعت (در دقیقه صفر) اجرا شود و مجموعه
- را پر کند **daily_data** اجرا شود و مجموعه (00:00 UTC) تابع روزانه باید هر روز
- بروید و بررسی کنید که اسناد جدید اضافه **daily_data** و **hourly_data** **Database > Collections** به Appwrite در داشبورد شده‌اند (حداقل 30 سند روزانه و 72 سند ساعتی برای عملکرد بهینه)

- بررسی تابع تحلیل هوش مصنوعی:

- اجرا شود (03:00 EEST, 09:00, 15:00, 21:00) تابع تحلیل باید هر 6 ساعت
- به کانال تلگرام بروید و بررسی کنید
- یک پیام متنی با تحلیل (شامل سیگنال، اهرم، تحلیل تکنیکال و بنیادی با حداکثر 1500 کاراکتر برای تحلیل بنیادی)
- RTL و قالب‌بندی Arial با تحلیل کامل، فونت (**.txt** نه **Aidin_AI_Analysis.docx** یک فایل
- "اگر سیگنال حرکت قیمتی کمتر از 400 دلار داشته باشد، پیام: "هیچ سیگنال معتبری با حرکت قیمتی حداقل 400 دلار تولید نشد"

- بررسی لاگ‌ها

- لاگ‌ها را بررسی کنید، **Logs > [تابع]** در **Functions**

- **Inserted record for [تاریخ]** یا **No missing hours to process**: برای تابع ساعتی
- **Inserted record for [تاریخ]** یا **No missing dates to process**: برای تابع روزانه
- با موفقیت به تلگرام ارسال شد، یا خطاهایی مانند **Word** دریافت شد، سند **Gemini/AvalAI** برای تابع تحلیل: تحلیل با موفقیت از **Word** خطا در تولید سند

- **Word** بررسی سند

- را از تلگرام دانلود کنید **.docx** فایل
- یا یک ویرایشگر سازگار بار کنید Microsoft Word با
- است. RTL و قالب‌بندی Arial، اطمینان حاصل کنید که شامل تحلیل کامل (شامل اهرم)، فونت



عیب‌یابی

- **docx**: به جای **txt**، دریافت فایل

- تابع تحلیل وجود دارند **package.json** در **form-data** و **docx** بررسی کنید که وابستگی‌های
- ناموفق **Word** برای ارسال سند **X** یا تلاش **Word** لاگ‌ها را بررسی کنید برای خطاهایی مانند **خطا در تولید سند**

- تابع را دوباره مستقر کنید



bash

```
npm install docx form-data
appwrite deploy function --functionId=<AIDIN_AI_FUNCTION_ID>
```

- داده‌های ساعتی/روزانه پر نمی‌شوند

- **Failed to fetch candle for** [تاریخ] لاگ‌ها را بررسی کنید برای خطاهایی مانند
- معتبر است و محدودیت درخواست‌ها را نقض نکرده‌اید **CRYPTOCOMPARE_API_KEY** اطمینان حاصل کنید که
- وجود دارند (**date** , **open** , **close** مانند) را بررسی کنید که ویژگی‌های موردنیاز **daily_data** و **hourly_data** مجموعه‌های

- تابع تحلیل اجرا نمی‌شود

- برنامه زمان‌بندی را بررسی کنید (**6/* * * 0** برای تحلیل، **0 * * * * 0** برای ساعتی، **0 * * * * 0** برای روزانه)
- اطمینان حاصل کنید که متغیرهای محیطی به درستی تنظیم شده‌اند
- لاگ‌ها را برای خطاهایی مانند متغیرهای محیطی ضروری وجود ندارد بررسی کنید

- پیام‌های تلگرام ارسال نمی‌شوند

- صحیح هستند **TELEGRAM_CHANNEL_ID** و **TELEGRAM_BOT_TOKEN** اطمینان حاصل کنید که
- بررسی کنید که ربات به‌عنوان مدیر در کانال تنظیم شده و مجوزهای لازم را دارد
- لاگ‌ها را برای خطاهایی مانند خطای ارسال به تلگرام بررسی کنید

- **API** محدودیت‌های

- (در لاگ‌ها **gemini_request_count** پیگیری با) حداکثر 500 درخواست روزانه Gemini)
- (جدید دریافت کنید API در صورت خطای 429، صبر کنید یا کلید) محدودیت درخواست‌ها را بررسی کنید CryptoCompare:

نکات اضافی

- است، تابع تحلیل در 03:00، 09:00، 15:00 (EEST (UTC+3) هستند. با توجه به اینکه UTC برنامه‌های زمان‌بندی به وقت **EEST** زمان‌بندی برنامه را به 21,3,9,15 0 * * * تغییر دهید، (EEST مثلاً 00:00) اجرا می‌شود. برای تغییر به زمان‌های خاص EEST و 21:00
- **hourly_data** و 72 ساعت داده در **daily_data** پشتیبانی از داده‌ها: برای عملکرد بهینه، اطمینان حاصل کنید که حداقل 30 روز داده در وجود دارد. توابع ساعتی و روزانه را ابتدا اجرا کنید تا داده‌ها پر شوند
- **APPWRITE_API_KEY** , **GEMINI_API_KEY** , **AVALAI_API_KEY** , **CRYPTOCOMPARE_API_KEY** (**API** امنیت: کلیدهای **TELEGRAM_BOT_TOKEN** را در جایی امن ذخیره کنید و از اشتراک‌گذاری آن‌ها خودداری کنید)
- عملکرد را Cron، تا قبل از انتظار برای اجرای (**Execute** > [تابع] > **Functions** در) تست دستی: می‌توانید توابع را به‌صورت دستی اجرا کنید بررسی کنید

پشتیبانی

اگر با مشکلی مواجه شدید، موارد زیر را بررسی کنید و اطلاعات را برای پشتیبانی آماده کنید:

- **Functions > Logs** لاگ‌های تابع در

- پیام‌های خطا در تلگرام



- متغیرهای محیطی تنظیم‌شده

- **ai_signals** , **daily_data** , **hourly_data** تعداد اسناد در مجموعه‌های

برای کمک بیشتر، این اطلاعات را با توسعه‌دهنده یا انجمن Appwrite به اشتراک بگذارید.

این راهنما تمام مراحل لازم برای راه‌اندازی سیستم را با جزئیات کامل پوشش می‌دهد. با دنبال کردن این مراحل، باید بتوانید توابع را مستقر کرده، داده‌ها را جمع‌آوری کنید، تحلیل‌ها را تولید کرده و به کانال تلگرام ارسال کنید. اگر سؤالی دارید یا نیاز به

کمک در عیب‌یابی دارید، جزئیات خطا یا لاگ‌ها را ارائه دهید!

