

Twitter Bot データベース定義

1. データベース要件

AWSのDynamoDBを使用してテーブルを管理する。

1.1. テーブル一覧

No.	テーブル名	説明	備考
1	twitterPosts	twitterに投稿する記事の一覧	
2	twitterHistoryPosts	1レコードずつ投稿していく	

1.2. テーブル定義

1.2.1. twitterPostsテーブル

No.	論理名	物理名	型	パーティションキー	制約	説明	備考
1	作成日	createdAt	String	○	UNICODE、NOT NULL	作成日をユニークとして管理する。 ms単位で計測。	
2	本文	text	String	-	NOT NULL	Twitterの本文を保持する。	
3	URL	url	String	-	-	URLを保持する。	
4	タグ	tag	String	-	-	タグを保持する。	
5	開始日	startPeriod	String	-	※開始日が入力されている場合、終了日が存在することをバックエンドで担保	投稿を開始したい期間を保持する。	
6	終了日	endPeriod	String	-	※終了日が入力されている場合、開始日が存在することをバックエンドで担保	投稿を終了したい期間を保持する。	

1.2.2. twitterHistoryPostsテーブル

No.	論理名	物理名	型	パーティションキー	制約	説明	備考
1	作成日	createdAt	String	○	UNICODE、NOT NULL	作成日をユニークとして管理する。 ms単位で計測。	
2	本文	text	String	-	NOT NULL	Twitterの本文を保持する。	
3	URL	url	String	-	-	URLを保持する。	
4	タグ	tag	String	-	-	タグを保持する。	
5	開始日	startPeriod	String	-	※開始日が入力されている場合、終了日が存在することをバックエンドで担保	投稿を開始したい期間を保持する。	
6	終了日	endPeriod	String	-	※終了日が入力されている場合、開始日が存在することをバックエンドで担保	投稿を終了したい期間を保持する。	
7	投稿済みフラグ	posted	String	○	UNICODE、NOT NULL	true、falseで管理、投稿が済んだフラグをtrueにする。	
8	登録日	registrationAt	String	○	NOT NULL	ms単位で計測。	

1.2.2.1. twitterHistoryPosts グローバルセカンダリインデックス

No.	論理名	物理名	型	パーティションキー	型	ソートキー	説明	備考
1	投稿日index	postedCreatedAtIndex	String	posted	String	createdAt	postedをパーティションキーとした検索用のテーブル	