Nerddy

PINS API

About the API

The PINs API allows apps to store small pieces of information on users on Nerddy's system. This information can be retrieved when the user makes requests to the app or when content from the app is loaded. This feature can be used to verify the user's identity (especially for apps that return user-specific output in a frame), associate other usernames/identities with the user, or create long-term sessions.

Authentication

• Requires appid and accesskey.

Limitations

• One PIN and five subids per user

• PIN/subid can contain letters in addition to numbers

• PIN/subid must not be longer than 250 characters

API URL

http://www.nerddy.com/beta/pins/v1

Supported Methods

POST

Request Format

Here is a sample request body:

{"appid":"xxxx","accesskey":"xxxx", "requestid":"xxxx","userid":"xxxx", "usersubid2":"9382g", "userpin":"123gdg"}

appid	Your app's parent (global) app ID
accesskey	Your parent app's access key
requestid	A random integer (required for reference)
userid	The ID of the user (varchar)
userpin	Specify data to store in the userpin variable for the user.
usersubid1, usersubid2, ..., usersubid5	Specify data to store in the usersubid variables.

Note: userpin and usersubid variables serve the same purpose and are essentially the same. In older API versions, userpin could only contain numbers, and usersubids could contain numbers and characters. As of version 1.0.0, userpin supports both numbers and characters.

Response Format

The response will return the status of the request.

{"requestid":"xxxx","result":"success","error":null}

Requesting PIN and Subids

Userpin and usersubid can be requested in the input processing stage and in the post-processing stage. In input processing, simply use return{ #!userpin!#; #!usersubid1!#; #!usersubid2!#;} to receive the associated values in the API communication stage. Parameters will typically not contain pound signs or exclamation marks when they are sent to your app. In post-processing, you can use #!userpin!#, #!usersubid1!#, etc. to print/echo the values on the user's front-end.