POSTing to e621 API

How-to

e621 requires you to use the `form` header for POSTing data:

```
let options = {
    uri: url,
    headers: {
        'User-Agent': 'my-user-agent',
        'content-type': 'application/x-www-form-urlencoded'
    },
    json: true,
    // this is your data you are POSTing to e621
    form: postObject
};
```

Typical Response

Responses from the e621 API typically look like this:

On Error

```
{ success: false, reason: 'Error Reason' }
```

On Success

```
{ success: true }
// ... Any additional information about the endpoint-specific POST results will also be returned
```

Logging In

If you just want to log in through the API, your headers should look something like this:

```
{
    uri: url,
    headers: {
        'User-Agent': 'my-user-agent',
        'content-type': 'application/x-www-form-urlencoded'
    },
    json: true,
    // this is your data you are POSTing to e621
    form: {
        login: "YOUR_E621_USER_NAME",
        password_hash: "YOUR_API_KEY"
    }
}
```

Notes

- You must provide the login `form` parameters on **every** POST call you make to the API. There is no per-session login system

- The e621 API does **NOT** use URL parameters when POSTing like it does for GET requests

- You **must** provide a meaningful User-Agent to the API or your POST request will be denied

- Sending POST messages to the e621 varies endpoint to endpoint but you need at provide two **required** properties in your `form` POST object:

```
postObject.login = "YOUR_E621_USER_NAME";
postObject.password_hash = "YOUR_API_KEY";
// any other parts of your postObj relating to the actual POST go here
```

- When POSTing with information, any properties that look like this: `post[source]` are to be used in the POST object like so:

```
formOptions = {
    "post[tags]": "your tags space_delimited",
    "post[source]": "source url where the image came from, can be blank but is required",
    "post[rating]": "rating string",
    "post[file]": "multipart form/file stream"
    //... your login info would need to go here as well
};
```