# Event Planner REST

Hierbei handelt es sich um eine in NodeJS geschriebene REST Schnittstelle für eine sehr simple Event/Party Verwaltung.

## Schritte zur Inbetriebnahme der API

Wichtig: Zur Inbetriebnahme muss NodeJS auf dem System installiert sein.

* Nach dem Entpacken der ZIP Datei mit der Konsole in das Verzeichnis wechseln
* Eingabe: *npm install*
* Start der Applikation mit dem Befehl: *node ./index.js*
* Die API ist unter der Adresse <http://localhost:3000/api/> erreichbar. Standardmäßig ist die API über den Port 3000 erreichbar. Der Port kann jedoch mithilfe eines Parameters angepasst werden (siehe Kapitel 2)

## Start-Parameter

Der Start der Applikation kann durch die Angabe von Parametern beeinflusst werden.

|  |  |  |
| --- | --- | --- |
| Parameter | Beschreibung | Beispiel |
| --cleanDB | Entfernt die Datenbanktabelle und somit alle vorhandenen Einträge | node ./index.js --cleanDB |
| --port x | Modifiziert den Port, unter welchem die REST API erreichbar ist. | node ./index.js --port 5080 |

## API Endpunkte

Nachfolgend finden Sie alle Endpunkte der REST API.

|  |
| --- |
| Fehlercodes |
| |  |  | | --- | --- | | Code | Bedeutung | | 11 | Falsche Parameter | | 20 | Interner Serverfehler | | | 21 | Objekt wurde nicht gefunden | |

|  |  |  |
| --- | --- | --- |
| /event | Holt Events aus der Datenbank ab einem bestimmten Datum | GET |
| |  |  | | --- | --- | | Parameters |  | | |  |  | | --- | --- | | Name | Beschreibung | | mindate (optional)  *(query)* | Gibt das Mindestdatum an, ab welchem Events aus der Datenbank gelesen werden sollen.  Format: YYYY-MM-DD z.B. 2018-01-05 | |  |  |  |  | | --- | --- | | Responses |  | | |  |  | | --- | --- | | Code | Beispiel | | 200  *(Success)* | [{  "id": 1,  "title":"Silvesterparty",  "expires":"2018-12-31"  }] | | 404 | {  "errorCode":21,  "message":"No entity found"  } | | 400 | {  "errorCode":400,  "message":"Bad date format. Must be YYYY-MM-DD"  } | |  | | | |

|  |  |  |
| --- | --- | --- |
| /event/count | Zählt Events aus der Datenbank ab einem bestimmten Datum | GET |
| |  |  | | --- | --- | | Parameters |  | | |  |  | | --- | --- | | Name | Beschreibung | | mindate (optional)  *(query)* | Gibt das Mindestdatum an, ab welchem Events in der Datenbank gezählt werden sollen.  Format: YYYY-MM-DD z.B. 2018-01-05 | |  |  |  |  | | --- | --- | | Responses |  | | |  |  | | --- | --- | | Code | Beispiel | | 200  *(Success)* | {  "count": 12,  } | | 500 | {  "errorCode":21,  "message":"No entity found"  } | | 400 | {  "errorCode":11,  "message":"Bad date format. Must be YYYY-MM-DD"  } | |  | | | |

|  |  |  |
| --- | --- | --- |
| /event/:id | Holt ein Event mit einer bestimmten ID ab | GET |
| |  |  | | --- | --- | | Parameters |  | | |  |  | | --- | --- | | Name | Beschreibung | | id  *(path)* | Gibt die ID des Events an, für welches Detailinformationen gelesen werden sollen.  Typ: integer | |  |  |  |  | | --- | --- | | Responses |  | | |  |  | | --- | --- | | Code | Beispiel | | 200  *(Success)* | {  "id": 1,  "title":"Silvesterparty",  "description":"Sekt und Häppchen für alle!",  "expires":"2018-12-31"  } | | 500 | {  "errorCode":21,  "message":"No entity found"  } | |  | | | |

|  |  |  |
| --- | --- | --- |
| /event/:id | Aktualisiert ein Event mit den übergebenen Daten | PUT |
| |  |  | | --- | --- | | Parameters |  | | |  |  | | --- | --- | | Name | Beschreibung | | *(body)* | {  "title":"Langweilige Silvesterparty",  "description":"Sekt und Häppchen sind aus!",  "expires":"2018-12-31"  } | |  |  |  |  | | --- | --- | | Responses |  | | |  |  | | --- | --- | | Code | Beispiel | | 200  *(Success)* | {} | | 404 | {  "errorCode":21,  "message":"Entity not found"  } | |  | | | |

|  |  |  |
| --- | --- | --- |
| /event/:id | Löscht ein Event anhand der übergebenen ID | DELETE |
| |  |  | | --- | --- | | Parameters |  | | |  |  | | --- | --- | | Name | Beschreibung | | id  *(path)* | Gibt die ID des Events an, welches gelöscht werden soll  Typ: integer | |  |  |  |  | | --- | --- | | Responses |  | | |  |  | | --- | --- | | Code | Beispiel | | 200  *(Success)* | {} | | 500 | {  "errorCode":21,  "message":"Entity not found"  } | |  | | | |

|  |  |  |
| --- | --- | --- |
| /event/create | Legt ein neues Event an | POST |
| |  |  | | --- | --- | | Parameters |  | | |  |  | | --- | --- | | Name | Beschreibung | | *(body)* | {  "title":"Alternative tolle Silvesterparty",  "description":"Weib, Wein und Gesang!",  "expires":"2018-12-31"  } | |  |  |  |  | | --- | --- | | Responses |  | | |  |  | | --- | --- | | Code | Beispiel | | 201  *(Success)* | {} | | 500 |  | |  | | | |