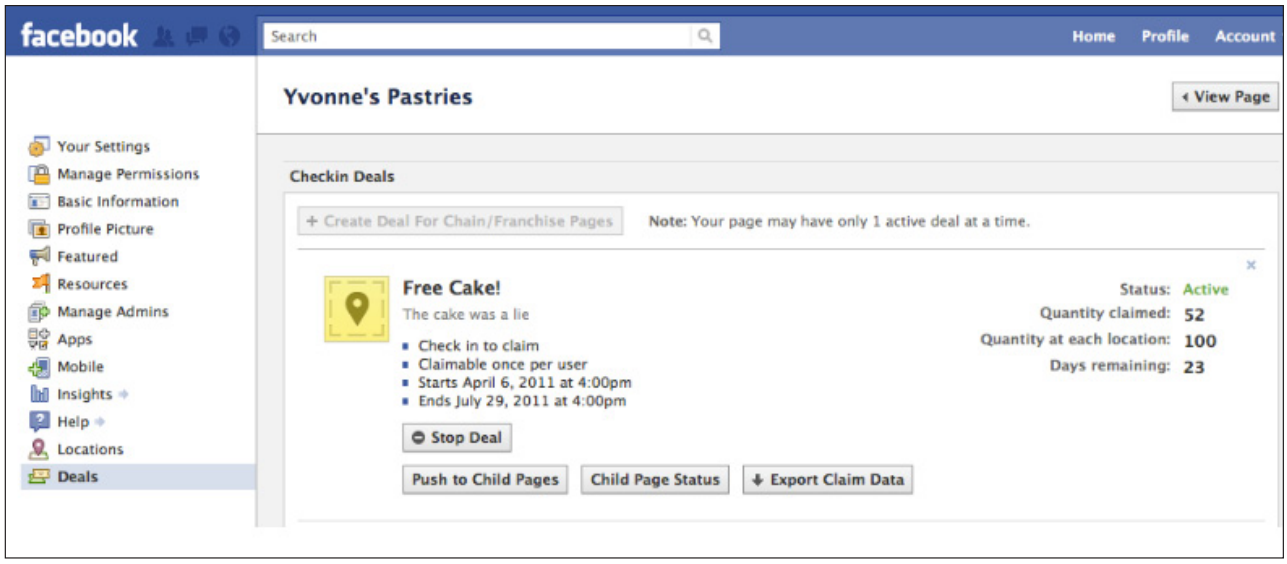
Check-in Deals API

The Check-in Deals API allows businesses with multiple store locations to create and manage Check-in Deals at scale without having to log into Facebook.

The new API saves time. Businesses are no longer restricted to run only one Check-in Deal across all their locations, and they don't have to manually create Check-in Deals on an individual basis. Now, they can now create many different Check-in Deals to run across specific locations.



Pictured above is the current tool on Facebook for creating Check-in Deals. This current system doesn't allow different deals to be assigned to groups of stores at scale.

Prerequisites for using the Check-in Deals API

- You must have worked with a Facebook representative to process your parent-child structure.
- An admin of the Page must grant the manage_pages permission and the Page's ID should be the subject of the Graph API call.
  Learn more at **https://developers.facebook.com/docs/reference/api/page**

Last updated July 2011

Recommendations for using the API

- **Create Prototype Check-in Deals.** These deals are basically a template that, once approved, can be quickly copied to children Places. This removes the need to create multiple individual Check-in Deals with the same information across multiple child Places.

- **Create regional Check-in Deals.** Create one prototype and then copy it to specific children Places. You'll be able to segment stores any way you like through your own Page/Place management applications, although you'll need to keep track of this segmentation on your own.

- **Some examples of Check-in Deal segmentation are:**
  – Regional Check-in Deals (from large geographies, to state, local regions, city, neighborhood, etc.)
  – Store type Check-in Deals (e.g. Yvonne's Pastries gets one deal, Yvonne's Super Pastries gets a different type of deal, and Yvonne's Subs gets no deal).
  – Driving foot traffic for testing purposes (i.e. run Check-in Deals only for markets that offer a certain product.)
  – A/B testing (e.g. see how specific Check-in Deals perform in similar test groups before expanding it more nationally.)

- **Reporting:** There is an opportunity for all the metrics and stats from the copied Check-in Deals to be aggregated into the Prototype Check-in Deal report. Please note that this functionality is forthcoming.

** While creating Check-in Deals is much easier through the API, all Check-in Deals must follow our guidelines and be approved by Facebook before going live.*

How to use the Check-in Deals API

The following steps must occur to use the API:

POST to graph.facebook.com/{**pageID**}/checkin_deals with the following parameters:

- **subtype** - required, values are: 1 for regular check-in deals, or 4 for charity check-in deals.
- **name** - required, must be nonempty. value is a string, 50 characters max. (e.g. 'deal summary' in regular deals).
- **description** - required, must be nonempty. value is a string, 100 characters max. (e.g. a 'thank you message' for charity deals, or a 'how to claim' instruction for regular deals.
- **friends_needed** - optional, if running a check-in deal that requires your friends to also check-in, the number needed to qualify for the deal (0 to 8, inclusive).
- **loyalty_frequency** - optional, if running a check-in deal that requires recurring checking, the number needed to qualify for the deal (0 to 20, inclusive).
- **start_time** - required, the start time of the deal, value is an ISO8601 datetime or UNIX timestamp.
- **end_time** - required, the end time of the deal, value is an ISO8601 datetime or UNIX timestamp.
- **max_inventory** - optional, the maximum number of deals that can be claimed, must be non-negative. defaults to 0 (e.g. no max).
- **can_repeat** - optional, is the deal repeatable or not, boolean. defaults to true.
- **is_prototype** - optional, is the deal a prototype deal or not, boolean. defaults to false.

Note: In order to propagate Check-in Deals to child places, **is_prototype** should be set to true. This call will return true or false depending on whether or not the deal was successfully propagated. The id of the promotion is then queued for approval. Once the prototype deal is reviewed and approved, you can then propagate the deal to child places:

POST to graph.facebook.com/{**promotionID**}/instances

- **child_place_id** - required, the id of the child place. The id must resolve to a Place that is a child of the page.