

Snack ver1.0

機能一覧

機能名:	メンバー新規登録
処理内容:	<ul style="list-style-type: none"> ・入力フォームから情報を受け取る ・name、pass、mail は必須。image は任意。 ・mail は重複がないかチェック。 ・エラーがなければ、メンバー情報をDB に保存。
必要データ:	①会員情報 ②mail
取得元:	①入力フォーム ②DB 「members.mail」
ユーザ操作:	<ul style="list-style-type: none"> ・フォームに入力 ・[register]をクリック

機能名:	ログイン
処理内容:	<ul style="list-style-type: none"> ・ログイン画面でセッションをまず消す ・入力フォームから mail と pass を受け取る。 ・DB から、mail と pass が一致するもののメンバー情報を取得 ・メンバーの id 他をセッションに登録し、マイページへ
必要データ:	①mail、pass ②メンバー情報
取得元:	①入力フォーム ②DB 「members」
ユーザ操作:	<ul style="list-style-type: none"> ・フォームに入力 ・[LOGIN]をクリック

機能名:	メンバー削除機能
処理内容:	<ul style="list-style-type: none"> ・ホーム画面で削除できるようにする。 ・メンバー情報を DB から削除。 ・そのメンバーが投稿したスナック情報を DB から削除。 ・そのメンバーイイネした like 情報を DB から削除。
必要データ:	①メンバーid
取得元:	①セッション
ユーザ操作:	<ul style="list-style-type: none"> ・削除リンクをクリック ・DELETE で完全に削除

機能名:	メンバー編集機能（名前と写真）
処理内容:	<ul style="list-style-type: none"> ・入力フォームから情報を受け取る ・name と image のみを変更できる ・image は指定しなければ、既存のものを再度保存する。 ・エラーがなければメンバー情報を DB にアップデート。
必要データ:	①メンバー情報（編集前） ②メンバー情報（編集後）
取得元:	①DB [members. Id,name,mail,image] ②入力フォーム
ユーザ操作:	<ul style="list-style-type: none"> ・フォームに入力 ・[EDIT]をクリック

機能名:	メンバー編集機能（パスワード）
処理内容:	<ul style="list-style-type: none"> ・入力フォームから情報を受け取る ・現在のパスワードを入力し、本人かチェック ・新しいパスワードを 2 回入力してもらい、DB にアップデート
必要データ:	<ul style="list-style-type: none"> ①パスワード（DB にある既存のもの） ②パスワード（入力された既存のもの） ③パスワード（入力された新しいもの）
ユーザ操作:	<ul style="list-style-type: none"> ・フォームに現在のパスワード入力 ・フォームに新しいパスワードを入力 ・[Change] をクリック

機能名:	Recommend 機能
処理内容:	<ul style="list-style-type: none"> ・お菓子を投稿し recommend して、DB に保存。 ・商品名、会社名、種類、国、コメント、キーワード、ホームページへの URL、商品写真を入力フォームに。 ・商品名、会社名、種類、国は必須。 ・コメント、キーワードは文字数制限を付ける。 ・URL は正規表現のバリデート。 ・商品写真は、拡張子を jpg,png,gif に限定。
必要データ:	①スナック情報
取得元:	①入力フォーム
ユーザ操作:	<ul style="list-style-type: none"> ・フォームに入力 ・[RECOMEND]をクリック

機能名:	検索機能
処理内容:	<ul style="list-style-type: none"> ・ キーワード、国指定、種類指定、順番指定でスナックを検索。 ・ 取得したスナックを,指定し他順番で一覧表示 ・ イイネ数、name、company、coment、recommender、image を表示。
必要データ:	①キーワード、国、種類、順番 ②スナック情報
取得元:	①入力フォーム ②DB 「snacks」
ユーザ操作:	<ul style="list-style-type: none"> ・ フォーム入力 & セレクト ・ [search] クリック

機能名:	お気に入り検索
処理内容:	<ul style="list-style-type: none"> ・ イイネしたものだけを表示。 ・ セッションの id と likes の member_id が一致する snack_id を取得。 ・ リレーションして Snack 情報を取得し表示。
必要データ:	①セッション id ②likes 情報 ③スナック情報
取得元:	①セッション ②DB 「likes.snack_id」 ③DB 「snacks」
ユーザ操作:	<ul style="list-style-type: none"> ・ [★] をクリック

機能名:	Recommender 検索
処理内容:	・あるメンバーが recommend した snack の一覧を表示。
必要データ:	①member_id (recommend した人の id) ②スナック情報
取得元:	①検索でスナック取得時に得た member_id を、url クエリで渡す。 ②DB [snacks]
ユーザ操作:	・ [Name] リンク クリック

機能名:	イイネ機能
処理内容:	<ul style="list-style-type: none"> ・ イイネをしてたら解除、してなければ登録。 ・ member_id と snack_id と一致するものを DB[likes] から検索。 ・ 一致するものがなければ likes に登録 & snacks.likes_cnt に+1。 ・ 一致するものがあれば likes から削除 & snacks_likes_cnt に-1。 ・ jquery の ajax を利用。
必要データ:	①セッション id (ログインしているメンバーの id) ②snack_id (検索時に取得したものの id) ③likes 情報 ④snacks.likes_cnt
取得元:	①セッション ②検索でスナック取得時に得た id を url クエリに渡す ③DB [likes] ④DB[snacks]
ユーザ操作:	・ [☆] リンクをクリック

機能名:	Suggest Snacks 機能
処理内容:	<ul style="list-style-type: none"> ・ログイン中のメンバーについて、イイネ傾向から自動でスナックを suggest する。 ・過去にイイネしたものの中で、国、種類、企業、recommender をランダムで選び、最も多いものを選択。 ・選択されたものの中で、まだイイネしていないものの中から、人気なものをランダムで2つ提示する。 ・ランダムでイイネ傾向に関係なく、人気なものを提示することもある。 ・ユーザーが好きそうなものを限定的に、多くのバリエーションで suggest するアルゴリズム。
必要データ:	<ul style="list-style-type: none"> ①セッション id ②likes 情報 ③スナック情報 (likes 傾向チェック) ④スナック情報(suggest)
取得元:	<ul style="list-style-type: none"> ①セッション ②DB[likes] (snack_id) ③DB[snacks] (type,country,company) ④DB[snacks]
ユーザ操作:	画面移動

機能名:	Snack 削除機能
処理内容:	<ul style="list-style-type: none"> ・セッション id と検索取得したスナックの member_id が一致してれば、削除できるようにする。 ・スナック情報を DB から削除 ・その LIKE 情報も DB から削除？
必要データ:	<ul style="list-style-type: none"> ①セッション id ②スナックの member_id ③スナックの id ④スナック情報
取得元:	<ul style="list-style-type: none"> ①セッション ②DB から取得した snacks.member_id ③検索取得時に得た snacks.id を url クエリに ④DB 「snacks」
ユーザ操作:	<ul style="list-style-type: none"> ・ [削除] リンククリックで削除画面に ・ [DELETE] クリックで削除

機能名:	Snack 編集機能
処理内容:	<ul style="list-style-type: none"> ・ DB から取得したスナック情報を表示。 ・ 入力フォームから情報を受け取る。 ・ エラーがなければ、DB にアップデート。 ・ キーワードは、既存のものに追加する形。 ・ image は選択されなければ、既存のものをそのまま再利用。
必要データ:	<ul style="list-style-type: none"> ①スナック情報 (DB) ②スナック情報(入力フォーム)
ユーザ操作:	<ul style="list-style-type: none"> ・ フォームに入力 ・ [REGISTER] を入力

機能名:	管理者用ログイン
処理内容:	<ul style="list-style-type: none"> ・セッション id=1（自分）のみ行けるようにする。 ・入力フォームで情報を受け取り、それによりログイン
必要データ:	<ul style="list-style-type: none"> ・管理者ログイン情報
取得元:	<ul style="list-style-type: none"> ・入力フォーム
ユーザ操作:	<ul style="list-style-type: none"> ・フォーム入力 ・[access]をクリック

機能名:	Snack 管理
処理内容:	<ul style="list-style-type: none"> ・各ユーザーに recommend された snack の表示、非表示を切り替える。 ・snacks テーブルの deletion の 0、1 をアップデート ・deletion0 で表示、deletion1 で非表示 ・deletion1 で非表示中だと、マイページの検索で表示されない。 ・Jquery の ajax を利用。
必要データ:	<ul style="list-style-type: none"> ・snacks_id
取得元:	<ul style="list-style-type: none"> ・検索取得した snack の id を hidden で渡す。
ユーザ操作:	<ul style="list-style-type: none"> ・[表示 or 非表示]をクリック

機能名:	メンバー管理
処理内容:	<ul style="list-style-type: none"> ・メンバーの制限、制限解除を切り替える。 ・members テーブルの deletion の 0、1 をアップデート ・deletion1 で制限されていない、deletion1 で制限されている。 ・deletion1 で制限中だと、ログインができない ・Jquery の ajax を利用。

必要データ:	・ members_id
取得元:	・ 検索取得した member の id を hidden で渡す。
ユーザ操作:	・ [制限 or 制限解除] をクリック