

Preferred Networks インターン選考 2017 コーディング課題フロントエンド分野

変更履歴

- 2017 年 5 月 12 日：初版

回答にあたっての注意

- サーバサイドは MacOS もしくは Linux で動作するようにしてください。
- クライアントサイドは HTML ベースで、Google Chrome もしくは Firefox で動作するようにしてください。
- サービスはインターネット上の他のサービスに依存せずに動作するようにしてください。
- 課題 1・2 はどちらもライブラリ・検索エンジン・データベースなどを使っても構いません。ただし、それらの導入手順は提出するドキュメントで示してください。
- プログラムは以下のいずれかの言語を利用してください。
- C, C++, Python, Ruby, Go, Java, JavaScript (AltJS を含む)

提出物

- 課題 1・2 を再現する全てのプログラム
- プログラムにはサーバサイド・クライアントサイドのソースコード以外に、検索エンジンやデータベースの設定ファイルやデータをインポートするために使うソースコードなども含まれます。
- サードパーティのライブラリは、導入方法をドキュメントに示せば、含めなくて構いません。
- 検索対象となるデータは含めないでください。
- 開発環境の構築に必要な手順を記載したドキュメント

提出方法

「事前課題の提出物」については Google drive にアップロードの上、共有 URL を下記の応募フォームに記入してください。アップロード手順は以下の URL をご参照ください。

- 応募フォーム：https://docs.google.com/a/preferred.jp/forms/d/e/1FAIpQLSd_zC_XT2dHM-yRO9WQ-YuRU0sx2HeQIep-NBoqMWpN_j8KNw/viewform
- アップロード手順：https://www.preferred-networks.jp/wp-content/uploads/2017/04/intern2017_GoogleUpload_3.pdf

問い合わせ

課題に関する質問などは intern2017@preferred.jp をお願いします（応募時と同一のメールアドレスです）

問題文

本課題では、外部のクラウドサービスを使わないウェブサービスのプロトタイプを作成してもらいます。提出物はソースコードと開発環境構築のドキュメントです（詳細は提出物を確認してください）。また、面接当日はデモやコードレビューが行われます。どちらも行える環境を準備してください。

課題 1

Wikipedia のダンプデータである `enwiki-latest-abstract.xml` のインクリメンタルサーチができるウェブサービスのプロトタイプを作成してください。

画面イメージ

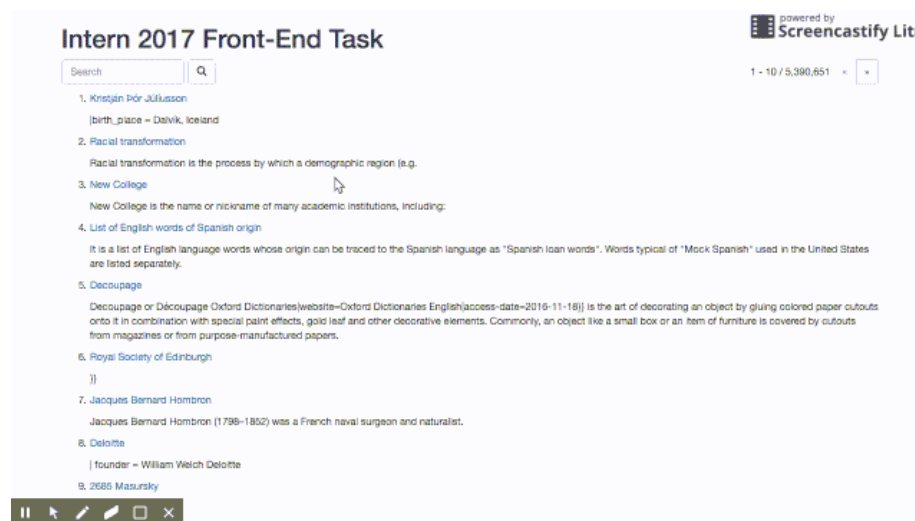


Figure 1:

この画像は `task1.gif` という名前と同梱されています。最低限の機能が満たされていれば、画面が一致する必要はありません。

要件

- プロトタイピングであるため、デモの範囲のアクセス数で動作する実装で構いません。
- 検索機能は以下の仕様を満たしてください。
- enwiki-latest-abstract.xml の **title** と **abstract** から検索をしてください。ただし、**title** に含まれる **Wikipedia:** という部分は除去してください。
- 小文字の検索クエリーの場合、大文字小文字は区別せずに検索してください。
- クエリーと完全一致するタイトルの記事はできる限り上位に表示してください。
- 検索クエリーが空の挙動は特に定めません。
- 検索 UI は以下の仕様を満たしてください。
- 入力ごとに検索結果が表示されるインクリメンタルサーチとしてください。
 - 表示はインクリメンタルサーチとして適切な速度で動作するようにしてください。
- 検索結果件数を表示してください。概数でも構いませんが、その場合は画面上で概数であることを示してください（例：*About 1,000,000*）。
- 検索結果は、項目ごとに **title** と **abstract** を表示してください。また、**title** には **url** へのリンクを新しいタブ・ウィンドウで開くように張ってください。
- 検索結果は 10 件ごとに表示し、それ以上はページネーションで閲覧できるようにしてください。
- 定まっていない挙動や UI は、UX が最大化されるようにしてください。
- 例えば、1 ページ目のページネーションの前のページのリンクは無効にする。

課題 2

課題 1 で作成した検索サービスに、検索しているクエリーをリアルタイムに全体に共有する機能（共有検索履歴と呼びます）を追加してください。

画面イメージ

この画像は task2.gif という名前で同梱されています。課題 1 の検索結果の右側に、共有検索履歴が追加されています。

最低限の機能が満たされていれば、画面が一致する必要はありません。

要件

- プロトタイピングであるため、デモの範囲で動作する実装で構いません。
- 共有検索履歴・UI は以下の仕様を満たしてください。
- 他のマシンで検索されたクエリーが、リアルタイム（数秒程度の遅れは許容されます）に共有されるようにしてください。
- 検索履歴は新しいものから最大 10 件が表示されます。
- 同じクエリーで検索が行われても、同じクエリーが複数表示はされません。もっとも新しいものだけが履歴として表示されます。
- 検索されたクエリーをクリックすると、検索フォームに入力され、即座に検索されるようにしてください。

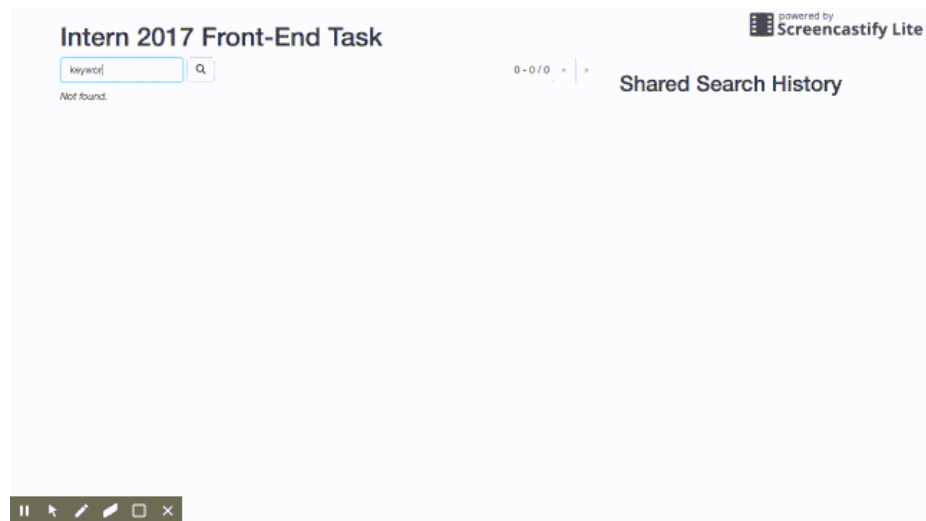


Figure 2:

- 課題1の検索サービスはインクリメンタルサーチであるため、入力中は履歴に登録されないように適当な工夫を行ってください。
- 例えば、クエリー入力後一定秒はクエリー履歴に送らない。
- 定まっていない挙動やUIは、UXが最大化されるようにしてください。