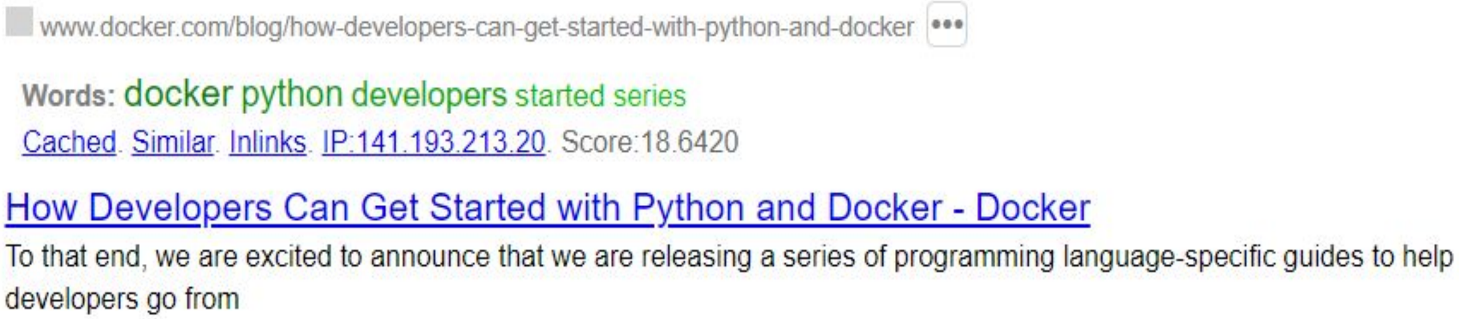
Understanding Yioop

UI and Search



www.docker.com/blog/how-developers-can-get-started-with-python-and-docker

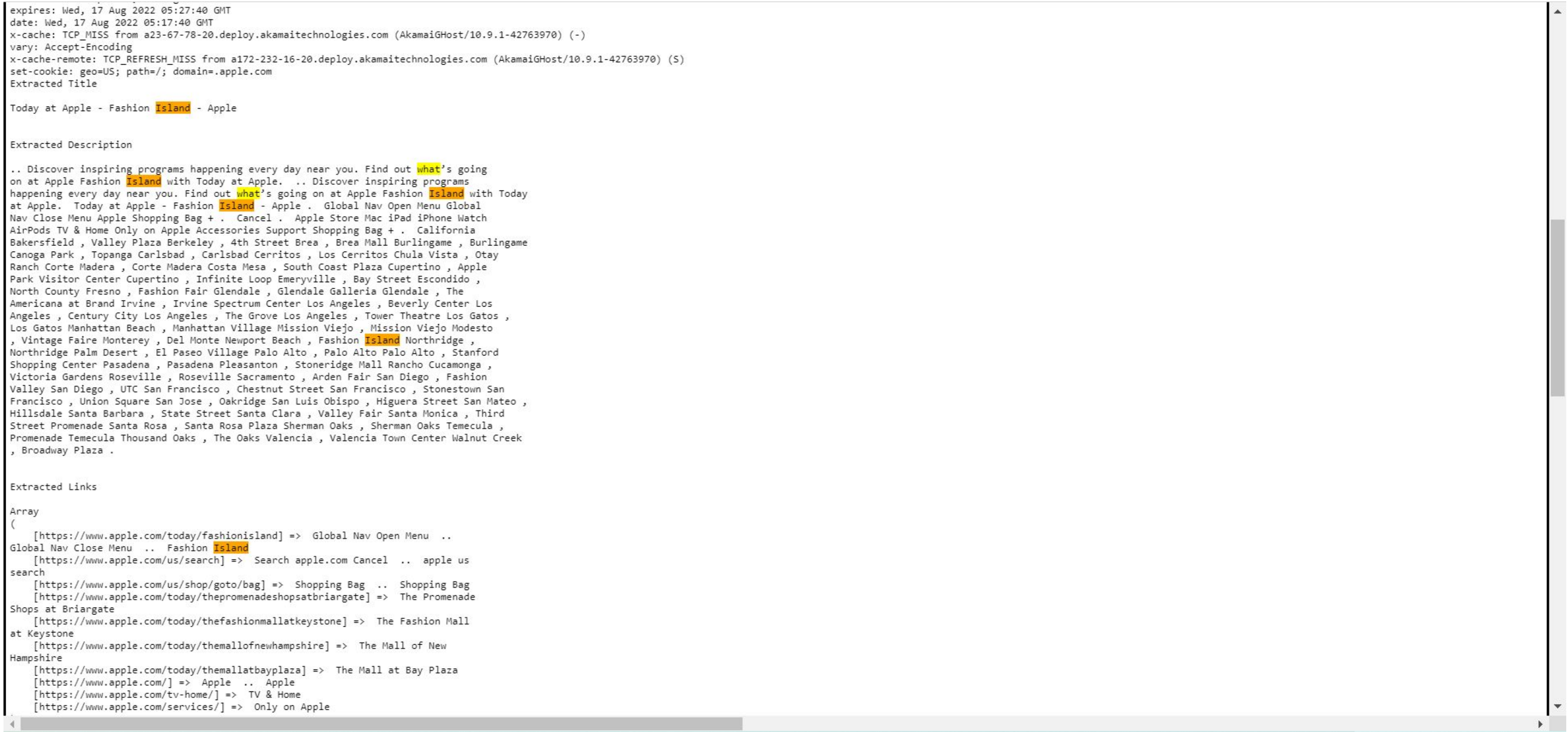
Words: docker python developers started series

Cached. Similar. Inlinks. IP:141.193.213.20. Score:18.6420

How Developers Can Get Started with Python and Docker - Docker

To that end, we are excited to announce that we are releasing a series of programming language-specific guides to help developers go from

- Simplified Url & favicon
- Word Cloud
- Cached Pages
- Search result similar to current doc (Search on top 5 relevant word)
- Incoming links
- Search score
- Docs with same IP address



```
expires: Wed, 17 Aug 2022 05:27:40 GMT
date: Wed, 17 Aug 2022 05:17:40 GMT
x-cache: TCP_MISS from a23-67-78-20.deploy.akamaitechnologies.com (AkamaiGHost/10.9.1-42763970) (-)
vary: Accept-Encoding
x-cache-remote: TCP_REFRESH_MISS from a172-232-16-20.deploy.akamaitechnologies.com (AkamaiGHost/10.9.1-42763970) (S)
set-cookie: geo=US; path=/; domain=.apple.com
Extracted Title

Today at Apple - Fashion Island - Apple


Extracted Description

.. Discover inspiring programs happening every day near you. Find out what’s going
on at Apple Fashion Island with Today at Apple.  .. Discover inspiring programs
happening every day near you. Find out what’s going on at Apple Fashion Island with Today
at Apple.  Today at Apple - Fashion Island - Apple .  Global Nav Open Menu Global
Nav Close Menu Apple Shopping Bag + .  Cancel .  Apple Store Mac iPad iPhone Watch
AirPods TV & Home Only on Apple Accessories Support Shopping Bag + .  California
Bakersfield , Valley Plaza Berkeley , 4th Street Brea , Brea Mall Burlingame , Burlingame
Canoga Park , Topanga Carlsbad , Carlsbad Cerritos , Los Cerritos Chula Vista , Otay
Ranch Corte Madera , Corte Madera Costa Mesa , South Coast Plaza Cupertino , Apple
Park Visitor Center Cupertino , Infinite Loop Emeryville , Bay Street Escondido ,
North County Fresno , Fashion Fair Glendale , Glendale Galleria Glendale , The
Americana at Brand Irvine , Irvine Spectrum Center Los Angeles , Beverly Center Los
Angeles , Century City Los Angeles , The Grove Los Angeles , Tower Theatre Los Gatos ,
Los Gatos Manhattan Beach , Manhattan Village Mission Viejo , Mission Viejo Modesto
, Vintage Faire Monterey , Del Monte Newport Beach , Fashion Island Northridge ,
Northridge Palm Desert , El Paseo Village Palo Alto , Palo Alto Palo Alto , Stanford
Shopping Center Pasadena , Pasadena Pleasanton , Stoneridge Mall Rancho Cucamonga ,
Victoria Gardens Roseville , Roseville Sacramento , Arden Fair San Diego , Fashion
Valley San Diego , UTC San Francisco , Chestnut Street San Francisco , Stonestown San
Francisco , Union Square San Jose , Oakridge San Luis Obispo , Higuera Street San Mateo ,
Hillsdale Santa Barbara , State Street Santa Clara , Valley Fair Santa Monica , Third
Street Promenade Santa Rosa , Santa Rosa Plaza Sherman Oaks , Sherman Oaks Temecula ,
Promenade Temecula Thousand Oaks , The Oaks Valencia , Valencia Town Center Walnut Creek
, Broadway Plaza .


Extracted Links

Array
(
    [https://www.apple.com/today/fashionisland] =>  Global Nav Open Menu  ..
Global Nav Close Menu  ..  Fashion Island
    [https://www.apple.com/us/search] =>  Search apple.com Cancel  ..  apple us
search
    [https://www.apple.com/us/shop/goto/bag] =>  Shopping Bag  ..  Shopping Bag
    [https://www.apple.com/today/thepromenadeshopsatbriargate] =>  The Promenade
Shops at Briargate
    [https://www.apple.com/today/thefashionmallatkeystone] =>  The Fashion Mall
at Keystone
    [https://www.apple.com/today/themallofnewhampshire] =>  The Mall of New
Hampshire
    [https://www.apple.com/today/themallatbayplaza] =>  The Mall at Bay Plaza
    [https://www.apple.com/] =>  Apple  ..  Apple
    [https://www.apple.com/tv-home/] =>  TV & Home
    [https://www.apple.com/services/] =>  Only on Apple
```

- Notice extracted title, summary, and links (SEO view of page)

Search Operators

- #num# : Used to show multiple query results. ‘num‘ represents number of search result for individual query
- query 1| query 2 : to search query **or** query 2
- “match query”: exact match search
- -notThisQuery: to search documents not containing ‘notThisQuery’
- related:url , cache:url , link:url , ip:ip_address: equivalent to clicking respective links of url/ip_address

Classifiers

- Sometimes text search is not enough to search document. Additional information about the pages can used to search.
- Classifiers adds label to meta words of crawled pages
- Classifiers can be created/trained manually by using ‘edit’ option of classifier

Page Indexing and search

- **Byte Range to Download**: number of bytes to download during crawl
- **Whole Page Cache**: downloaded bytes are stored in archives
- **Summarizers**: summarizer to use to summarize the text (Basic, Centroid, Centroid Weighted, and Graph Based)
- Basic summarizer uses tag scraping and distance from top
- Centroid methods splits document into sentences and calculates average sentences. Use this to determine which sentence to add to add to the summary
- Graph based uses PageRank style approach where weighted adjacency matrix is between sentences is calculated using a notion of similarity between two sentences.

Page Indexing and search

- **Max Page Summary Length**: number of bytes of summary which is sent to the queue server
- **Suffix Phrases:** Whether to extract suffix phrases from document summaries at crawl time and whether to use suffix phrases at search time.
- **Allow Page Recrawl After**: Number of days Yioop keep tracks of downloaded URL. After these days, bloom filters are reset to allow recrawling
- **Page File Types**: types of pages to crawl
- **Classifiers and Rankers**: Classifiers and Rankers to use to classify and rank pages

Page Indexing and search

- Control search result elements displayed attributes
- Give weightage to search ranking factor - title, description, and links
- Yioop searches docs till it find certain number of qualifying docs
- Sorts the qualifying docs and return top 10 results
- In multi-queue-server setting, query is sent simultaneously to queue servers and results are aggregated
- Name server requests alpha * (minimum result group)/(number of queue server) docs. Server Alpha controls the number alpha

Search Ranking Mechanism

- Without operators, Yioop uses conjunctive queries i.e. tries to find doc with all search terms
- Search score is based on 3 main scores -
    - Doc Rank (DR)
    - Relevance (Rel)
    - Proximity (Prox)
- Only scans n number of documents till timeout
- This is based on assumption first n doc contain top 10 results.
- Assumption is true as Yioop indexes docs based on doc rank. Rel and Prox do not affect results drastically

Crawl time ranking

- A Name server, which acts as an overall coordinator for the crawl, and which is responsible for starting and stopping the crawl.
- One or more Queue Servers, each of which maintain a priority queue of what to download next.
- One or more Fetchers, which actually download pages, and do initial page processing.
- Hash of hostname of each seed url is computed and assigned to queue server. All urls having same hostname will be assigned to same server
- Fetchers get active crawl info of queue server from NameServer. Fetcher downloads pages from queued urls

Fetchers and their Effect on Search Ranking

- Downloads batch of 100 pages at a time
- Downloads only PAGE_RANGE_REQUEST bytes by sending in request header
- After batch is downloaded performs -
    - Choose Page processor based on mime type
    - Extract summary using page processor
    - Apply indexing plugin
    - Run classifiers
    - deduplication by hashing after removing tags and non words
    - Prune number of links extracted to max limit
    - User defined rule on extracted summary
    - Keep summaries and complete page (if configured) in memory until downloaded complete schedule or SEEN_URLS_BEFORE_UPDATE_SCHEDULER many have been downloaded. At this point, summaries and caches are shipped off to the appropriate queue server in a process we'll describe later

Indexing

- Indexer:
  - Responsible for indexing
  - Adds Index Data file information to the active partition and periodically launches a DictionaryUpdater sub-sub process.
- DictionaryUpdater:
  - When the active partition get full, a new partition is started
  - Builds an inverted index for the old active partition and adds the result to the overall index

IndexDocumentBundle

- **Documents**: Sequence of file pairs called partitions: partition_SOME_INTEGER.ix, partition_SOME_INTEGER.txt.gz Each partition typically stores around 100,000 documents on an 8GB RAM machine
- Given partition_SOME_INTEGER.txt.gz file contains a sequence of gzip-compressed document summary, gzip-compressed document objects.
- The file partition_SOME_INTEGER.ix contains a sequence of records of the format (doc_id, offset in txt.gz file to summary, offset in txt.gz file to document, length of document object)
- pdb_parameters.txt which contains information about compression and record formats used, the max size in bytes for a partition, the maximum number of record for a partition, etc

IndexDocumentBundle

- positions_doc_map:
  - Consists of a sequence of integer numbered folders corresponding to partitions in the previously described documents folder
  - Each folder except the last folder contains three files: a doc_map file, a positions file, and a postings file. The last folder has in addition a last_entries file.
  - doc_map file consists of a sequence of tuple pairs doc_id => position_score_list where position_score_list is a list of pairs (position, score) . The first such (position, score) is the offset to the document in the txt.gz partition file, followed by an overall score for the document in the partition.
  - (position, score) 's after the first score are position term positions within the representative document, scores for the terms from this position and the previous. So a second tuple pair (10, 0.5) would indicate term locations 0, 1, 2, ..., 10 should each be weighted 0.5 when determining how important a term having one of these location is.
  - After these set of pairs the position_score_list has additional pair (0, user_score) for the scores of the document with respect to each classifier being used for the crawl.

IndexDocumentBundle

- ‘positions’ is a binary file used to store for each term found amongst a partition's worth of documents, the locations of the term within each document that it occurred in.
- Such a list is stored using a gamma-code for the first value, followed by a Rice-code of a difference list of the remaining values.

IndexDocumentBundle

- The postings file has two similar formats, one for the partition new documents are being added to and for other partition.
- For the new document partition, it contains an inverted index for that partition.
- Inverted index: A sorted-and-grouped-by-term sequence of tuple pairs. (term_id => posting_list_for_term) , where posting_list_for_term , consists of, for each document the term appears in, a tuple (index of document in doc_map file, frequency of term in document, offset of terms position list in positions file, length of positions file entry).
- For all other partitions, the format is almost the same, execpt term_id's are not in the file as they are already stored in the B+-tree dictionary entry.

IndexDocumentBundle

- last_entries file is used for record keeping for each term to be able to output postings correctly. For a given term it consists of a triple (term_id, last_doc_index, last_offset, num_occurrences)

IndexDocumentBundle

- Dictionary:
    - Implemented B+ Tree using folder structure. Folder represents internal node. file represents leaf node.
    - key-value pair: term_id -> posting_list_for_term
    - term_id -> sorted list of partition info records for term_id. A partition info record is a tuple (partition number, number of docs term appeared in for the partition, total number of occurrences of term in partition, offset into postings file where postings for partition can be found, length of posting data).

IndexDocumentBundle

- next_partition.txt
  - Contains a single integer indicating the next partition that could be added to the dictionary that been yet.
- archive_info.txt
  - This file contains information about the creation time of the archive, the crawl parameters used to obtain the data stored in the archive, and the archive version format.