

Quick reference documentation for the dredger

This documentation is intended to clarify the doubts which could have arisen given the nature of the script. It's a very simple script, so this intends to answer the why rather than the "how". As per usual, it's written in an ISO-9001 manner so it's feasible for everyone.

Context: The idea with this script is to be able to get our Tappedout decks (<https://tappedout.net/>) into a database where we will be able to organize them. Also, the database isn't made from the point of view of a standard collector but rather from a Commander Player, where each card is intended to be only once, no matter the edition, and if a card belongs to two decks it's proxied on either of them hence in the collection only a single copy is present.

Vocabulary:

Beforehand, I'd like to state that the card used for examples is this: <https://scryfall.com/card/9ed/100/storm-crow>

Card: this refers to a Magic The Gathering Card. Currently we only have on it's definition its name, its quantity, its set (full name) and its status. This is to be expanded in the future

Deck: A deck is a set of cards with play fuctionality

Card name: The name of the card. E.g., the name of our card is Storm Crow

Set: The set the card belongs to. It can be in abbreviation or full name. Our card is from Ninth Edition (full name), also called 9ED (abbreviation)

Status: The status of the card. The statuses are as follows

Deck:{deckname}	It's on the deck {deckname}
Incoming	It has been ordered and it's on its way here
Collection	It's unused on the collection
Ordered	The order for the card has been placed yet it still has to arrive
Pending	The card must be ordered

Quick reference documentation for the dredger

Structures:

Import file:

Quantity [space] Card name [space] ({set(abb.)} Ost{status}st0

E.g.:

1x Canopy Vista (BFZ) Ostcollectionst0

Which is a single Canopy Vista from the set whose abbreviation is BFZ and its current status is "Collection"

Set dictionary file:

{set(abb.)}. {set(full)}

E.g.:

RtR.Return to Ravnica

Which as a dictionary sets "RtR" as a key for the value "Return of Ravnica"

Table (current)

CREATE TABLE public.cartas

```
(
  id integer NOT NULL DEFAULT nextval('cartas_id_seq'::regclass),
  quantity smallint NOT NULL,
  name character varying(90) COLLATE pg_catalog."default" NOT NULL,
  set character varying(50) COLLATE pg_catalog."default" NOT NULL,
  status character varying(30) COLLATE pg_catalog."default" NOT NULL,
  CONSTRAINT cartas_pkey PRIMARY KEY (id),
  CONSTRAINT cartas_quantity_check CHECK (quantity > 0)
)
```

Quick reference documentation for the dredger

Conclusion:

Right now, both the database and the script are both functional yet there's still room to grow. At the moment the focus is on creating a script to set the status automatically to a whole file, to normalize set names (E.g.: when a card is from a special edition out of the collector numbers for the edition it's edition'll appear as {XXX:YYY} instead of {XXX}) and database-side to include missing fields and update the script to include them