Hasura // Dec 5, 2020

yaksha

1 Flipbook

A flip book is a book with a series of pictures that vary gradually
from one page to the next, so that when the pages are turned
rapidly,
the pictures appear to animate by simulating motion or some other
change.

Design a language for describing flipbooks and implement a
compiler
for this language that can convert a flipbook description into a print
able pdf (or a video).

1.1 Example

Let’s say we want to create a flipbook which shows the ‘Human Life Span’ by displaying images of childhood, adolescence, young adulthood, adulthood and old age, one after the other. For this flipbook, our language just needs a single primitive - displaying an image on a range of pages. The flipbook may then be defined as follows:

```
01 05 child.jpg // Display a child's picture from
pages 1 to 5 06 10 adolescent.jpg
11 15 young-adulthood.jpg
16 20 adulthood.jpg
21 25 old-age.jpg
```

An accompanying compiler should be able to generate a viewable format from the flipbook definition. For example

```
fc human_life_span.flip -o human_life_span.pdf
```

where fc is a compiler of the above language. Instead of pdf, you can choose to generate a video instead, say a gif or an mp4 file, whichever is convenient.

1.2 Notes

1. A flipbook language would require more primitives to describe scenarios which are complex than the above example. For example, for the same 'Human Life Span' filpbook, if we were to display the images one **next** to the other, we would need some way to position an image on the page which the above language does not support. Similarly which primitives are needed to create a flipbook to show an Apple falling on Newton's head with your language?
2. The problem is rather open-ended. You can use a language of your choice and any libraries that you need. Feel free to get creative and show off and take however long you'd like. Just let us know how much time you were able to spend on it. Doing the task over one or more sessions is totally ok.
3. Apart from your programming maturity and the way you organize your codebase, we're looking for your ability to think as a builder of tools. The design of your language, its current scope, its future scope and extensibility. Feel free to design a complete language and only implement a core subset to keep a check on the amount of time you can spend and dedicate to the task

1.3 Submission

1. Given our developer facing nature and our open-source roots, we'd love it if you could make sure your work is easy for us to evaluate with decent docs, and instructions to try it out. Have an example or two flipbooks that can be converted to pdf/video by running your compiler.

2. Push your code to a private Github repository and add 3 accounts as collaborators:
    a. accounts@hasura.io
    b. vamshi@hasura.io OR 0x777
    c. tanmaig@hasura.io OR coco98

3. Record a small screencast of you giving a demo and taking us over the language design.