Proposal

to

Ebury Press

for a

Voice Activated Cooking Assistant

Author: Ross Payne

Date: Jan 2019

What is the offer?

The use of an Alexa Cooking App (also called an Amazon Skill) for no upfront cost, that is designed to compliment the purchase of each Ebury Press cook book by delivering the recipe content in new and exciting ways.

If you believe, after reading this document, that the App may have something to offer Ebury Press then I can arrange for members of your staff to evaluate the App at their leisure.

As a minimum please read the first two pages followed by the screen shot demonstratiing the App in Appendix B.

What does the App do?

The App provides a virtual cooking assistant for Ebury Press cook books, using a combination of Amazon Alexa voice assistant and intelligent display, with emphasis on the personalisation of recipe content.

The App facilitates searching for recipes across a users library of Ebury Press books, by *recipe name, ingredient* or *keyword*. Ingredient quantities can be scaled *on-the-fly* to suit a cook's requirement and the ingredient listing displayed in the familiar Ebury Press format of the respective cook book. Clear and simple cooking instructions can be recited to the cook at their prompting for the complete recipe or selected parts of it.

Why Ebury Press?

While the App provides a more convenient and error free way to view and execute recipes when in the kitchen, it is useless without great recipes. I could source content from the internet, however recipes from unknown sources is not what I and other foodies want. We demand recipes from trusted authorities like those published by Ebury Press, which is why we purchase your cook books in the first place.

So why an app?

There is no doubt about the enduringly popularity of cook books, as evident in their consistently stratospheric sales. Anyone with a modicum of interest in food loves to slowly browse through a good cook book given a moment, enticed by the photography, inspired by the recipe names and transported by the passion in their hands. The satisfying experience continues as you delve into a recipe. You study the ingredients and read the prose with a growing intent.

The difficulties starts, I find, when you move from the comfort of the reading space (both metaphorical and physical) into the kitchen. In brief, cooking while mentally tracking a list of ingredients, quantities and units and constantly reviewing a recipe text for instructions, with hands covered in flour and butter, is stressful and error prone, or at least for some of us, maybe even the majority, given the mountainous volume of cook books sold and the relative paucity of self-declared competent cooks.

What we need is a better way to deliver recipe content while in the kitchen. A *hands free, voice activated virtual cook assistant,* delivered by Alexa through my App, represents the best way to communicate a recipe to a cook. Succinct voice commands, when properly designed and delivered, will reduce the errors, stress and preparation time to as close to zero as possible.

The result is an App design that has deconstructed a traditional recipe format into some fifty unique data descriptors, some sourced directly from the original recipe, others inferred and a good portion that are entirely new. When fully populated a recipe may contain between 250 to 650 specific instances of "data" depending on its complexity. The benefit of all this data are recipe formats that can be reimagined and presented to the cook in new and exciting ways starting with today's voice assistants through to interactive intelligent displays of the future.

Now, instead of dashing between mixer and book with annoying regularity, the cook can request Alexa to recite a comprehensive set of cooking instructions. Each instruction designed to do one simple task, typically with one ingredient e.g. "*measure 30 grams of caster sugar into a small bowl*". The fundamental benefit of a verbal assistant, when properly implemented, increases the confidence of an unskilled or average cook to tackle any recipe, no matter how complex or unfamiliar, with vastly reduced apprehension and preparation. The App however, can do much more than just recite instructions. Read on.

Requirements for the App

The App is designed for so called *intelligent displays*, specifically the Amazon *Echo Show*[1]. Support for Google's intelligent displays is a definite future development.

The market uptake of these devices, in America and even more so in Australia, has been spectacular not the least because of their low cost and the general appeal of a virtual assistant that facilitates a hands free verbal interface to everyday applications, such as this App.

These products are strategic to Amazon and Google and represent how these companies view applications being consumed now and into the future. The appeal of these products will only increase as the versatility and power of the devices increase, the speech improves and more applications become available. I dare say, over the next five years, intelligent displays located in the kitchen will be regarded as *de-rigueur,* as much as the iPad is in the family room and bed-side table today.

The second and third requirement requires the App user to have purchased an Ebury Press cook book and registered the book on the Ebury Press website[2]. This will require a some sort of key, I imagine, that is entered as proof of their purchase. Thereafter, the App will display the *registered book or books* on the home page. All recipe searches will be limited to these books.

Benefits to Ebury Press

The proposal, at a minimum, offers Ebury Press an opportunity to evaluate an innovative way to deliver recipe content to the kitchen at relatively low risk. If pursued to the end, Ebury Press will be in a position to market their cook books as "***Alexa Enabled***" which will appeal to a rapidly growing portion of the population.

More importantly though, the cooking skill offers Ebury Press a uniquely intimate marketing channel direct to the customer in the comfort of their kitchen, and on a more frequent basis than a cursory browse of a books at the local book shop or the quick read of the latest cook book review simply cannot.

As an example, the Apps powerful ***recipe search*** feature, which is currently limited to searching in books purchased by the customer, could be extended to search all Ebury Press cook books, or perhaps, a more limited selection chosen by the customer. While the App would not allow access to the details of a recipe sourced from a book they have not purchased, the App could display tantilising snippets of the recipe that may stimulate a future book purchase.

[1] I also expect it will support the yet to be released Echo Show 5, a much cheaper Echo Show with half the screen size.

[2] The Ebury Press website currently does not have this functionality. This is part of the Apps supporting infrastructure that needs to be developed.

Features of the App

The cook book will always be the preferred way to peruse recipes and find inspiration but once a recipe is chosen, this Alexa App may well be the preferred way for a good number of cooks to execute a recipe in the kitchen using the App's rich set of features.

User Registration

The App does not register a user as such. Rather it is aware of two identifies, the *email address* that is assigned to the *Echo Show* device when it is first setup by the owner, and a *device id,* that is associated with each device. As far as the App is concerned the *device id* represents the user and it is the *device id* that is used in all communication with the database, while the *email address* is used only once to find the books that have been registered against that email address from the Ebury Press website. Thereafter separate program (yet to be developed) will synchronise the App's database with new book registrations.

On the Ebury Press side, the website currently allows interested members of the public to register their email address. *The website would need to extend this to allow a user to also register an Ebury Press cook book.* This is further documented in the section*: ?? to Ebury Press.*

Recipe Search

The Alexa App has the facility to search for recipes based on *keyword(s)*[3] (e.g. pasta, fish, flourless cake, vegetarian, desert etc), *ingredient(s)*[4] or the *full recipe name* or *part of the recipe name.* For example, "*search ricotta pudding*" will return all recipes that have ricotta and pudding in the recipe name, "*search tarragon ravioli*" will display all recipes that have tarragon and ravioli in the name (in any order) and "*search pasta*" will list all recipes that have the keyword *pasta* associated with the recipe or pasta in the recipe name. To search for all chocolate cake recipes requires the cook to say "*search chocolate cake*" or "*search chocolate cakes*".

The verbal command "*search*" can be replaced by any word or phrase that Ebury Press would prefer, e.g. "*get me* <blah> *recipes*", "*find* <blah>", "*list* <blah> *recipes"* etc.

By default the App will search across the Ebury Press books registered to the user/device. Optionally, the user may ***open*** one of the books (say "*open book Plenty*") prior to conducting a search in which case the search will be restricted to recipes in that book. The user can ***close*** the book at any point (say "*close book*" or simply "*close*") and all subsequent searches will be across their registered books.

Searching uses an index that is generated when a recipe is loaded into the database. The index is generated based on the following recipe naming syntax adopted by Ebury Press:

[a [[,b] and] d] [<type>] subject [with [type] e [and f]]

where:

a,b,d,e,f are ingredients.
subject is the food category e.g. tart, cake, name-of-fish, pudding, etc
type is a qualifier for subject or ingredient

In the case of recipe:

rice pudding with roasted rhubarb and tarragon

[3] *keywords and phrases* are defined as part the recipe data. There is no limit to the number of keywords that can be assigned to a recipe.

[4] *as* used in the recipe name only.

Twenty two index entries are generated and a further eleven virtual entries exist as the query logic will reorder the words in the search phrase as it searches for recipes e.g. “rhubarb tarragon” is defined in the index and “tarragon rhubarb” is its virtual entry[5].

List Ingredients

Once a recipe has been selected, the user will be given the option to list the ingredient quantities.

The format of the listing follows exactly the format adopted in the associated cook book.

If a recipe is divided into *parts* (e.g. filling, topping, garnish) then the listing will similarly be broken into parts.

Currently the ingredient listing is only displayed to the user, not recited, simply because I don’t see any value in this type of interaction, however this can be incorporated very easily.

I am currently investigating ways to send the ingredients listing to either the user’s printer or to the accompanying Alexa app[6] on the users mobile, so they can consult the ingredient list when shopping.

Scale a Recipe

This maybe a somewhat controversial feature but it comes from a personal requirement to reduce the ingredient quantities for some recipes. Consequently the App enables the user to verbally adjust **all** quantities to some fraction of the original using a single command, “*scale <some percentage>*”. This feature will only allow the user to scale down i.e. reduce the quantities. The *list ingredients* command will display all the ingredient quantities adjusted to their new values. Similarly, Alex will take into account the adjusted quantities when reciting the recipe instructions.

A scaling factor can be applied any number of times, including scaling back to the original quantities by saying “*scale reset*” or “*scale one hundred percent*”.

Importantly the App keeps the ingredient format consistent between adjustments. For example, if the current quantity is shown as a fraction, e.g. “1/2 cup”, and the scaling factor is 0.5 then the value displayed will be “1/4 cup” not “0.25 cup”. One litre, displayed as “1l”, under a 0.33 scaling will be displayed as “330ml”, rather than 0.33l, where rounding has adjusted it to the nearest 5ml[7]. However, for values less than “1/4 tsp” the App will replace the quantity with “a pinch”, which is not ideal but what can you do. Scaling a recipe will never be as accurate as the original, but in the vast majority of cases it proves more than satisfactory.

To give a visual context to the user, the header portion of the display shows the current scaling factor at all times.

The data model permits this feature to be disabled for an individual recipe if desired.

An alternative to directly scaling the ingredients is to adjust quantities based on a container size, which will be discussed next.

Scale by Container Size

[5] from a cost and performance perspective there is a delicate balance between storage size of the index and CPU and IO of the query logic. The preference has been to increase the index size so the CPU and IO can be reduced.

[6] not to be confused with this App that runs solely on the Echo Show. The accompanying Alexa mobile app is an Amazon product that provides administrative functions to Alexa intelligent devices.

[7] rounding applies to any value over 20

Some recipes, typically cakes, size the ingredient quantities to suite a particular container dimension. The "*size*" verbal command, enables a user to adjust all the ingredient quantities to suit the size of **their container**, provided the dimension is less than the original recipe container.

For example, a cake recipe may specify a 30 cm round cake tin, however the cook may only have a 26 cm round cake tin available, in which case the cook can say "*size twenty six*" and the App will multiple all quantities by 0.7511 (26**2/30**2). As with the *scale* command, the ingredient listing and recipe instructions will make use of the adjusted values.

This option can be disabled for an individual recipe if desired.

List Containers and Utensils.

Cooking *containers* (bowls, cake tin, trays, etc) their *size* (small, medium, large) as well as cooking *utensils* (whisk, oven, electric mixer, rolling pin, grater etc) form a part of the recipe data, particularly relevant in the graph version of a recipe (more about that later).

The verbal command, "*list containers*" displays the number of containers[8] and their size that a recipe requires. It also list the utensils used by the recipe.

Recite Recipe Instructions

The principal use case of the App is to recite recipe instructions. It should be understood that instructions in this context, are not simply sentences recited from the text in a book, as this content does not lend itself to fail safe execution of a recipe, in my experience.

A single sentence from a book recipe may refer to many ingredients, often simply comma separated (up to five I have seen), requiring many separate activities across multiple containers to completely fulfil the intent of the sentence. A *verbal instruction* on the other hand, should represent a single *non-divisible activity* (within reason) applying to a *single ingredient or mixture*. Keeping an instruction as simple as possible reduces the chance of the cook making an error. Consequently one sentence in the original recipe text can easily expand to six or more verbal instructions.

Each verbal instruction should contain all the information required to complete it, such as the *ingredient quantity, size, container dimensions* or *oven temperature* when relevant. The user should not need to consult any other data source to fulfil a verbal instruction.

Alexa will start reciting an instruction with the command "*list instructions*". Alexa will wait for up to twenty minutes for the user to respond with either "*next*", "*say again*" or "*previous*"[9]. Previous will get Alexa to recite the previous instruction. The user can also respond with "*list containers*" or "*list ingredients*" should they need to review the containers or ingredients for any reason.

Recipe Parts

Some recipes fall naturally into separate components which in turn are represented by their own recipe. It follows that "parts" have their own *ingredient listing* and *set of cooking instructions*. *SWEET* and *NOPI* make extensive use of recipe parts as indeed do most recipe books.

[8] the data model permits container usage to be tracked throughout a recipe, consequently it can determine if a container can be reused, in which case it will only count it once and show the number of containers as *n-m e.g. 2-3*, meaning 2 containers if reused otherwise 3.

[9] If the user does not respond after twenty minutes Alexa will exit the App, in which case the user will need to relaunch the App - using the *launch phrase*.

The data model has a very flexible approach to dividing a recipe into parts. A recipe can be divided by *ingredient*, such as “Sause”, “Topping” or by *time* , such as instructions that can be carried out on the “day-before” and those that can be performed, “on-the-day”. A single recipe can be configured with both types of aggregation.

In addition, a recipe can be divided into separate *threads of execution*, that can be performed concurrently if desired. Threads are discussed in the next section.

To list the recipe parts the cook says “*list parts*”. This will list the recipe parts by name that have been aggregated by *ingredients* and *time*. The user then selects a part and the Alexa will start reciting the associated instructions when prompted. The selected part becomes the *active part* until changed. The user can say at any point thereafter “*list ingredients*” to show the ingredient listing and then resume cooking by saying *“list instructions*” and the App will return to where they left off. When the cook finishes executing the instructions for a part the App will return the cook to the parts listing so they can select the next part to compete.

If there is no *active part* and the recipe is configured with parts, “*list instructions*” will display the list of parts as if the user had said “*list parts*”.

Concurrent Recipe Instructions

Complex recipes may require the cook to manage two or more concurrent set of instructions.

For example, a *set of instructions* representing *stream A of a recipe*, is put on hold while the ingredients are cooking in the oven, at which point the cook starts *stream B* of the recipe only to be interrupted fifteen minutes later by the oven bell ring alerting him or her that stream *A* has finished cooking. The cook has the option to either complete *stream A* and return to *stream B* or complete some steps from *stream A*, come back to *stream B* and complete the remainder of stream A. There is no correct way to complete a multi-stream recipe and is entirely up to the cook.

How can the App manage this scenario?

While the data model accommodates configuring “*streams of execution*” into a recipe the App has no control over the cook’s timeline and the decisions they make to manage these *concurrent streams*. A confident cook may decide to perform the threads concurrently in the shortest time possible, while a less confident cook may decide to perform the streams consecutively. The App should not dictate one way of the other must be completely subservient to the different approaches cooks will take to complete a recipe.

The App provides a verbal command, ***resume,*** to switch between the multiple streams of a recipe[10]. The cook is aware that a recipe has multiple stream when the screen display splits into two or more virtual screens, each showing the instructions relevant to its stream. The active stream is highlighted in *green.* When it is appropriate the cook can swap to the non-active stream by saying “*resume*”, at which point that stream is now the active one and Alex will start reciting instructions from that stream when prompted. At any point the cook can swap between streams by saying “*resume*” again. When a stream completes the cook can say “next” and Alexa will automatically resume from the last incomplete stream. If may sound complicated but it is a very effective way to handle complex recipe instructions.

With a combination of the display showing *active* and *non-active* threads and a verbal command to select a thread the App maintains the cooks flexibility to follow a complex multi-threaded recipes in any particular order.

Two Interfaces

[10] only two concurrent threads are supported at this stage.

The App supports both a *graphic user interface* (GUI) via the touch screen of the Echo Show, and *voice commands* via Alexa, to navigate through the various features of the app. The App is intended however, to be used solely via voice commands, as this permits not only hands free access to each feature, but also direct access to said feature, whereas the GUI is less efficient, as it must follow a hierarchy of screens to gain access.

Session State

The App maintains a cooks session state for up to three days. This enables the cook to relaunch the App within this period and resume from exactly where they left off previously.

Subliminal Alexa

Producing a natural sounding Alexa was critical to the subliminal experience of using the App.

Simply getting Alexa to recite a recipe instruction that is displayed on the *Echo Show* screen was adequate, but experience showed that if you gave Alexa a more appropriately worded and punctuated instruction, it would vastly improve the oral perception[11]. This immediately translated to a better understanding of the instruction on first listening.

The source of the problem resulted from two clashing requirements. One, to display a recipe instruction with the minimum of verbiage so it's easier to read at a glance. The other, to get Alexa to recite the instruction with the verbiage, as that makes Alexa sound more human.

For example, a simple instruction is displayed on the display as:

“add 3/4 tsp of bicarbonate of soda to a small bowl”

This is suitably succinct and conveys all the information accurately and abbreviates where expected.

However this does not work quite so well for Alexa. To sound more human Alex needs to recite:

*“**now** add 3/4 **of a teaspoon,** of bicarbonate of soda**,** to a small bowl”*

Note the extra words and punctuation - all designed to make Alexa sound less robotic and a bit more human.

Without going into too much detail the App's data model stores two versions of an instruction, one to be displayed the other for Alexa,

```
"txt" : { "S" : "add {measure} of {ingrd} to {addtoc}“ },
"say" : { "S" : "now add {measure} of {ingrd}, to {addtoc}" },
```

The recipe designer is now free to tailor an instruction specifically for Alexa, adding extra words and punctuation where necessary. The other part of the solution is programmatic, where tokens like *{measure},* would expand differently for Alexa.

Making Alexa sound more human contributed significantly to the appeal and effectiveness of the App, so much so, that I feel justified in including it as a feature.

How much will it cost?

There is **no** upfront fee.

[11] The data model also supports using MMS for Alexa

I am prepared to give Ebury Press full use of the code and App for free. The kudos and the opportunities that may bring me in the future is sufficient. I will however charge a small monthly fee for the design, which will be tied to the App's popularity.

The largest component of the cost, is the opportunity cost to Ebury Press to complete most of the outstanding tasks required to finalise the App. These are documented in the table below. The time to complete these tasks should not be underestimated. There could be at least 6 months worth of effort across multiple disciplines. The App has the foundations to be good but only when the outstanding tasks are fully completed can its full potential be realised. I expect the best approach to control costs is to do a limited rollout across one or two books initially and only in the UK region. This will reduce the data requirements and allow us to gauge the popularity and usage of the App before committing more resources.

In addition, there will be a monthly cost for the consumption of AWS (Amazon Web Service) Cloud resources. The App runs entirely in the AWS Cloud so there are no other computing costs.

The AWS fee is based on a *pay-for-what-you-use* model[12]*,* so if no one uses the App there will be no fee. As my monthly fee will be tied to the AWS fee both costs are effectively tied to the popularity of the App. If the App is not particularly popular then you can expect a charge of **tens of pounds to maybe a couple of hundred pounds a month**. If the App is widely popular then this may edge towards a thousand pounds a month. For the first couple of years I would expect it will be on the lower side of these figures.

My monthly fee will also cover code maintenance for the first twelve months and for monitoring of the Apps performance and operational metrics.

Any ***significant*** enhancements that you deem important I will carry out for a small fixed fee, proportional to the complexity of the enhancement, remembering that it is in my best interest to make the App as good and as popular as possible. *The App does the basics and perhaps a little more, so I encourage your input to make it better.*

You can **terminate** the use of the App at any point. There are no commitments with either AWS or myself, and terminating the App will immediately terminate all future monthly fees.

The anticipated costs to Ebury Press for finalising the App are documented in the following table.

Task	**Who is Responsible**	**What needs to be done?**
Integration	Ebury Press	Determine how to register a book purchase on your website in such a way that it cannot be registered more than once. I imagine this will require a printed **key** that uniquely identifies the book, that is either printed in the book or receipt?
	Ebury Press	Once the above design has been resolved, your website will need to be enhanced to enable book registration against the owners email address. Currently your website facilitates registering a persons email address only.
	Ebury Press Myself	A simple program needs to be written, or your website enhanced, to copy new book registrations from the website's database to the App's database in AWS.

[12] there is also an alternative model that requires a small fixed monthly charge that may turn out to be cheaper depending on the popularity of the app. As there are no commitments to either plan, it is possible to change the model as circumstances change to minimise the monthly costs.

Add Visual Branding	Ebury Press Myself	The App represents a blank canvas as far as branding and other visuals are concerned. Little to no regard has been spent on the visual design as this is best left to the experts. Consequently, I would appreciate working with the talented graphic designers at Ebury Press to create an App that is visually appealing, pleasant to use and is stamped with the Ebury Press sense of design. The App has eight screens currently.
Design and Load Recipes	Ebury Press Myself	This represents a significant amount of work owing to the number of cook books. The initial release should be restricted to a subset of books to reduce the work and get the App out to the public as soon as possible. It takes on average between two thirds to one day to deconstruct a book recipe into the data elements required by the App, load the data into the database, trial the recipe via the App and humanise Alex's responses. Simple recipes may only take an hour while more complex recipes may take over a day and a half. Changes to the data model and code may be required as new data elements are discovered in books that have not previously examined. Ebury Press staff can be trained to do the vast majority of this work.
Amazon Certification	Myself	Amazon must certify the App prior to its public release.

New Role

The App has significant data requirements beyond the recipe content found in cook books. The person responsible for orchestrating the recipe data, has the role of "recipe designer". They must breakdown a recipe, into its constituents parts namely, *ingredients, set of instructions (both displayed and verbal), containers, utensils, measures, alternative measures, units, specifying elapsed time for a task, defining searchable keywords* and specifying the *flow of ingredients between containers.* In addition they are responsible for breaking down a recipe into *parts* based on *ingredients, instructions/time* and/or *threads.* Once the data is defined and loaded into the database they are then tasked with the testing the recipe using the App for accuracy and fluency of Alexa's instructions.

The Data Model

The objective of the design from the outset, has been to define all aspects of a recipe as a single JSON document, from which can be projected many different *formats* such as, text, speech and graphical and *data content,* from simple ingredient listings to more complex multithreaded recipes and graphic displays.

```
                         recipe data
                    (single JSON document)
                              |
                     [program orchestrates]
                              |
                              V
        _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
        |              |               |              |              |
   Ingredient     Containers     Instructions.      Parts/        Pictorial
    Listing.        Listing     (display+verbal)   Streams    Representation*
```

Recipes from *SWEET, River Cafe* where used to model the base requirements of the data model. In addition, there were a number of my own requirements, namely, a recipe must be able to be represented in a *pictorial form*[13], and all quantities be scalable in real-time. This necessitated all *quantities*, *measures* and *units* be normalised and stored in a semi-parsed form enabling programmatic support for scaling-on-the-fly. As a by-product this will enable all recipes to support **internationalisation**[14] with relatively little effort in the future.

The App Architecture

The App runs entirely in Amazon's AWS cloud and is completely *serverless,* from Alexa, at the top of the stack, to the database at the bottom.

Layer	Description	AWS Service
Display	Alexa Presentation Language (APL)	-
Alexa Voice Services	Interprets voice commands from the user and recites text sent to it from the program logic	Alexa
Program Logic	Responds to the commands sent to it from the Alexa Voice Service. Orchestrates data from the database to satisfy the request and sends it to the Alex Voice Service and displays it on the device using the Alexa Presentation Language.	Lambda (JS + Go)
Database	Contains session state and recipe data	DynamoDB

All levels of the infrastructure will automatically scale to cope with demand and will support up to 1000 concurrent requests a second so there is a lot of growth built in.

Future architectures may make use of dedicated resources, to optimise costs, but until then the *serverless* model represents the cheapest and most reliable approach.

Recipe as a Graph

The pictorial representation described here is not part of the current App, but is intended to give a glimpse of what may be possible in future versions. The pictorial representation demands much larger displays than is currently available from todays intelligent devices. This immediately presents a problem for kitchens that are short on space but in the future who knows what form factor displays will take. Will they be embedded in cupboard doors, will they be the cupboard door, or refrigerator door or portions of a wall.

The pictorial representation of a recipe is not intended to replace the time honoured prose with attached list of ingredients. As I have said earlier, the cook book is the preferred way to consume recipe content for inspiration and contemplation. The pictorial representation on the other hand, is intended to be used in the kitchen as a visual guide to executing a recipe that goes beyond the mere reciting of instructions.

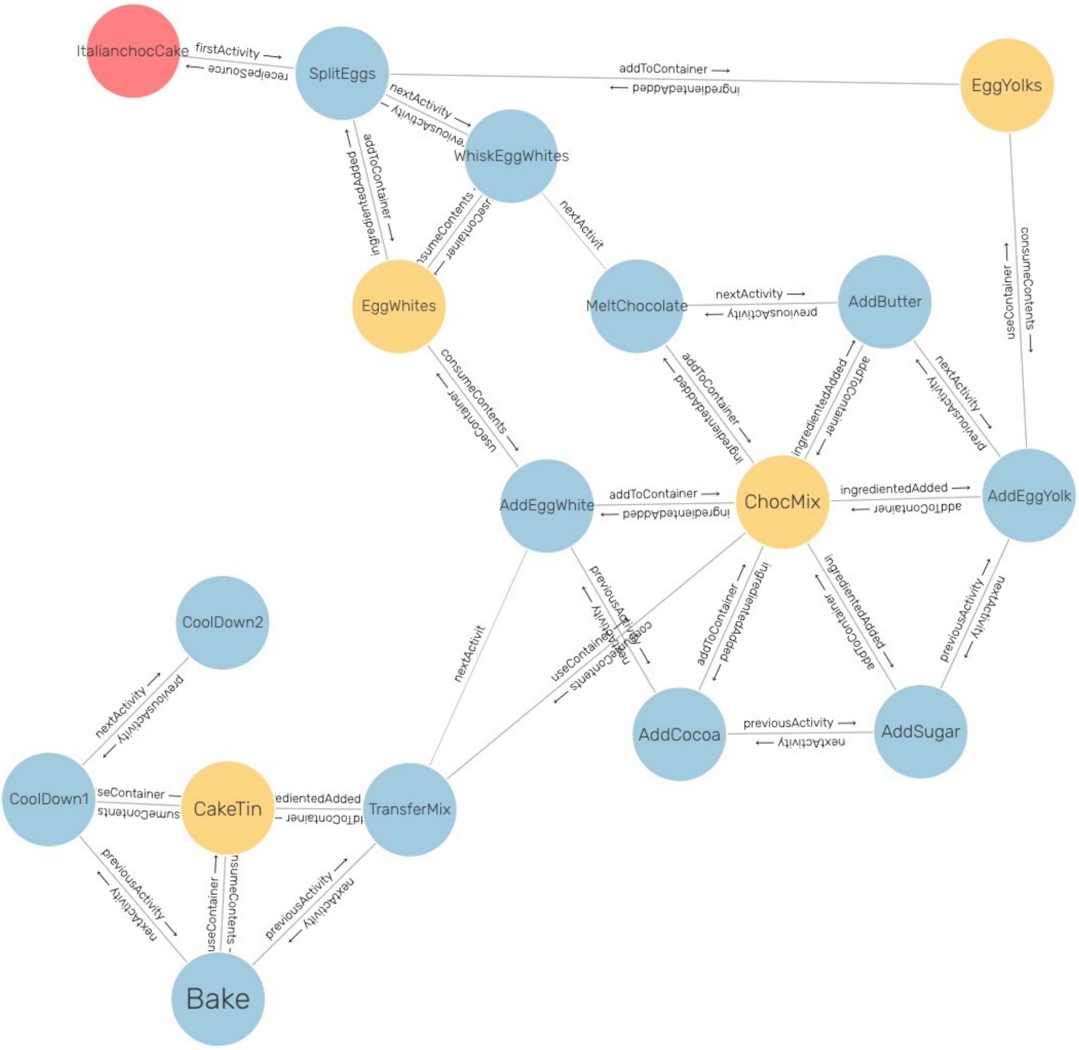
Depicted in the diagram below is the recipe, *Italian Chocolate Cake* from River Cafe shown in pictorial form, or as a *graph,* to give it its more formal name.

The blue nodes represent *activities,* where each activity performs some action on an ingredient, typically adding it to a *container*, shown in yellow. Each *activity* equates to a single instruction in the current App. As the cook progresses through the recipe the *current activity* would be

[13] specifically in the form of a *directed acyclic graph*

[14] support both metric and imperial units from a single copy of a recipe.

highlighted, maybe turning from blue to green, providing a visual context of where the cook is up to in the recipe.



The yellow nodes, as stated, represent *containers.* An activity typically adds an ingredient to a container (see *MeltChocolate, AddButter* activities), but it may perform some action on the contents of a container (see *WhiskEggWhites)*, or consume the entire contents of a container by transferring it to another container (see *TransferMix)*.

The lines (or edges in graph speak) are directed, meaning they point from one node to another (see arrows) in time execution order. Essentially they represent the *flow of activities or ingredients across containers.* Each time the cook says “*next*” the App progresses along the *nextActivity edge* to the attached activity.

From the above recipe you can hopefully see at a glance that there are four containers and about a dozen instructions and immediately get a sense of how complex or simple the recipe is. In this case it is simple, as the half of activities surround a single container. You can also see that most activities add an ingredient to a container where only a few activities transfer the contents of one container to another.

Appendix A: Schema design

An excerpt from a recipe JSON document (765 lines in total). Note liberal use of attributes/tokens.

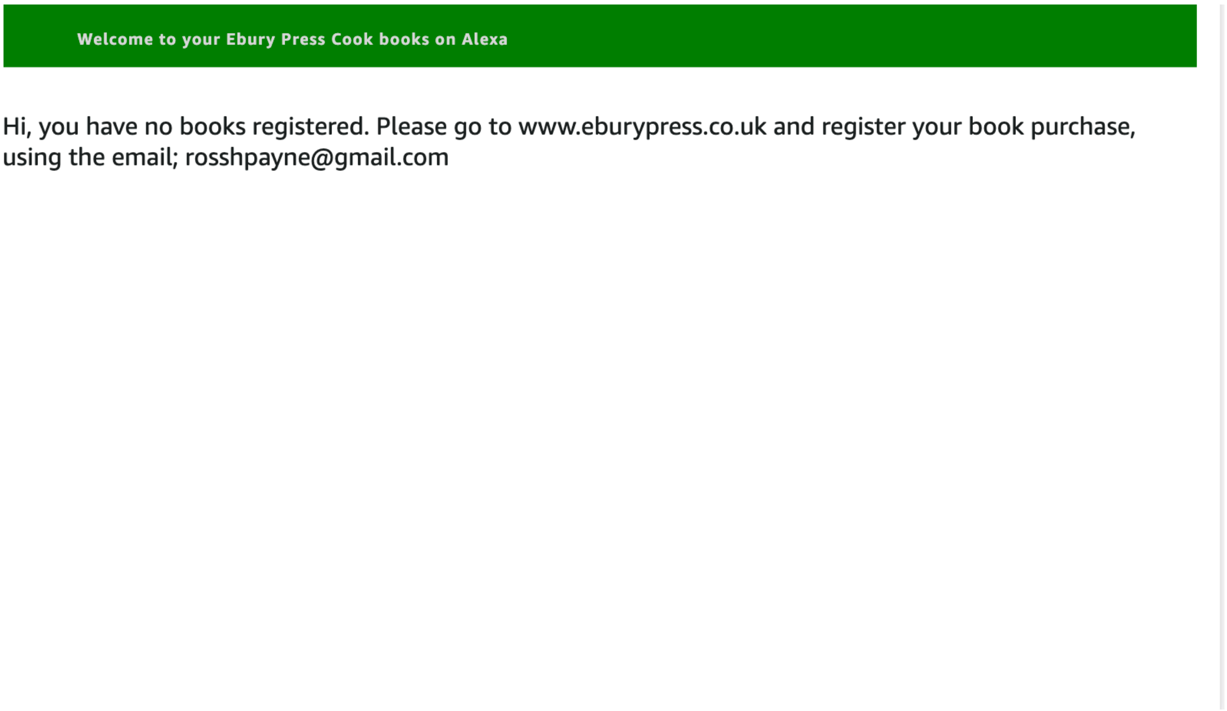
```
          "Item": {
                "PKey" : { "S": "A-20-4" },
                "SortK" : { "N": "180"},
                "part" :  {"S": "ck"},
                "label" : { "S": "eggs"},
                "coord" : { "L" : [ {"N": "13" }, {"N": "23" } ]  },
                "ingrd" : { "S" : "eggs" },
                "measure" : { "M" : {
                                      "qty" : { "S" : "3" } ,
                                      "size":{"S": "large"}}
                },
                "prep" : { "L" : [
                                {"M" : {
                                  "txt" : { "S" : "set aside {measure} {ingrd} to come to room temperature" },
                                  "say" : { "S" : "set aside {measure} {ingrd} to come to room temperature " },
                                  "time" : { "N" : "20" },
                                  "tplus" : { "N" : "0" },
                                  "unit" : { "S" : "sec" }
                                   }
                                }]
                                },
                "task" : { "L" : [
                                {"M" : {
                                  "txt" : { "S" : "add the {measure} {ingrd} one at a time blending well between each" },
                                  "say" : { "S" : "add the {measure} {ingrd} one at a time blending well between each" },
                                  "time" : { "N" : "2" },
                                  "tplus" : { "N" : "0" },
                                  "unit" : { "S" : "min" },
                                  "addToC" : {"SS" : ["EMixerBowl"] }
                                   }
                                }
                                ]
                }
        "Item": {
                "PKey" : { "S": "A-20-4" },
                "SortK" : { "N": "190"},
                "part" :  {"S": "ck"},
                "label" : { "S": "vanilla"},
                "coord" : { "L" : [ {"N": "13" }, {"N": "23" } ]  },
                "ingrd" : { "S" : "vanilla extract" },
                "measure" : { "M" : {
                                      "qty" : { "S" : "1" } ,
                                      "unit": {"S": "tsp"}
                }},
                "prep" : { "L" : [
                                {"M" : {
                                  "txt" : { "S" : "preheat {useD} to {temp} " },
                                  "say" : { "S" : "preheat {useD} to {temp}" },
                                  "time" : { "N" : "15" },
                                  "tplus" : { "N" : "0" },
                                  "unit" : { "S" : "min" },
                                  "useD" : { "M" : {"type" : {"S": "oven" },
                                                   "temp" : {"S": "200/180" },
                                                   "unit" : {"S": "C" },
                                                   "set"  : {"S": "Gas Mark 6" }
                                                   }}}
                                },
                                 {"M" : {
                                  "txt" : { "S" : "To bake the cake we, we make use of a {useC.form} if that is what you have.
Eitherway, just make sure it is a springform or a tin with a removable base." },
                                  "say" : { "S" : "To bake the cake we, we make use of a {useC.form} if that is what you have.
Eitherway, just make sure it is a springform or a tin with a removable base." },
                                  "useC": {"SS" : ["caketin"] }}
                                 },
                                {"M" : {
                                  "txt" : { "S" : "Lightly grease and line the {useC} with baking parchment and set aside" },
                                  "say" : { "S" : "Lightly grease and line the {useC} with baking parchment and set aside" },
                                  "useC": {"SS" : ["caketin"] }}
                                 }
                            ]
                },
                "task" : { "L" : [
                                {"M" : {
                                  "txt" : { "S" : "add the {ingrd} to the {addtoc} and blend " },
                                  "say" : { "S" : “now add the {ingrd} to the {addtoc}, and blend" },
                                  "time" : { "N" : "30" },
                                  "tplus" : { "N" : "0" },
                                  "unit" : { "S" : "sec" },
                                  "addToC" : {"SS" : ["EMixerBowl"] }
                                   }
                                },
                                {"M" : {
                                  "txt" : { "S" : "add the dry ingredients slowly and beat until combined well" },
                                  "say" : { "S" : "add the dry ingredients slowly, and beat until combined well" },
                                  "time" : { "N" : "30" },
                                  "tplus" : { "N" : "0" },
                                  "unit" : { "S" : "sec" },
                                  "sourceC": {"SS" : ["flourBowl"] },
                                  "addToC" : {"SS" : ["EMixerBowl"] }
                                   }
                                },
```

Appendix B - Screen shots from a cooking session.

What follows are screen shots from the Alexa Simulator of a cooking scenario designed to demonstrate the salient features of the App.

The development database contains approximately fifteen recipes, from *SWEET* and *River Cafe,* in various states of completion.

It should be emphasised that the screen designs are very plan as they contain no graphic content or branding material. You will have to imagine the Ebury Press stamp of design on each screen I'm afraid.

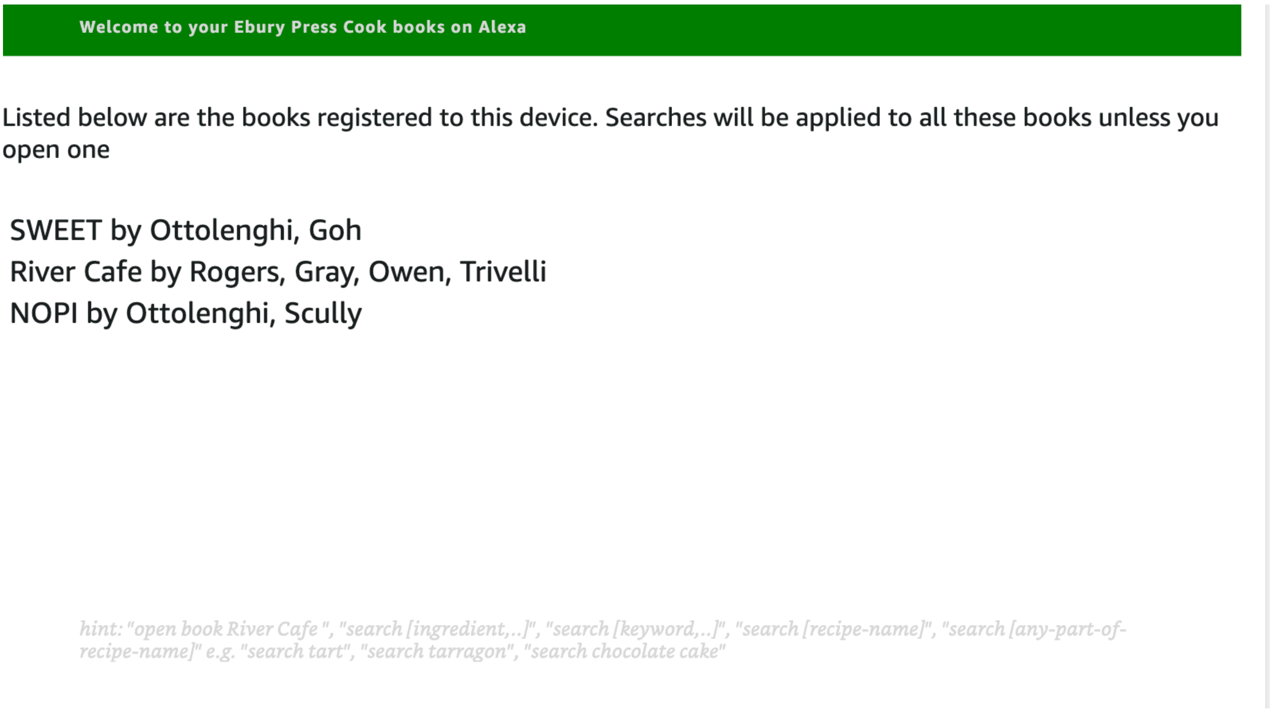


Welcome to your Ebury Press Cook books on Alexa

Hi, you have no books registered. Please go to www.eburypress.co.uk and register your book purchase, using the email; rosshpayne@gmail.com

*Cook Said: "**alexa <app launch phrase>**"*

The above screen shot shows the HOME screen when there are no books registered to the displayed email address. The email address is assigned to the device (Echo Show) at setup time and the App sources the value from the Amazon cloud, provided the user has given permission to do so, otherwise the App will ask for permission before retrieving the email address.
The cook at this point should purchase an Ebury Press cook book and register it on the website and then relaunch the App.



Welcome to your Ebury Press Cook books on Alexa

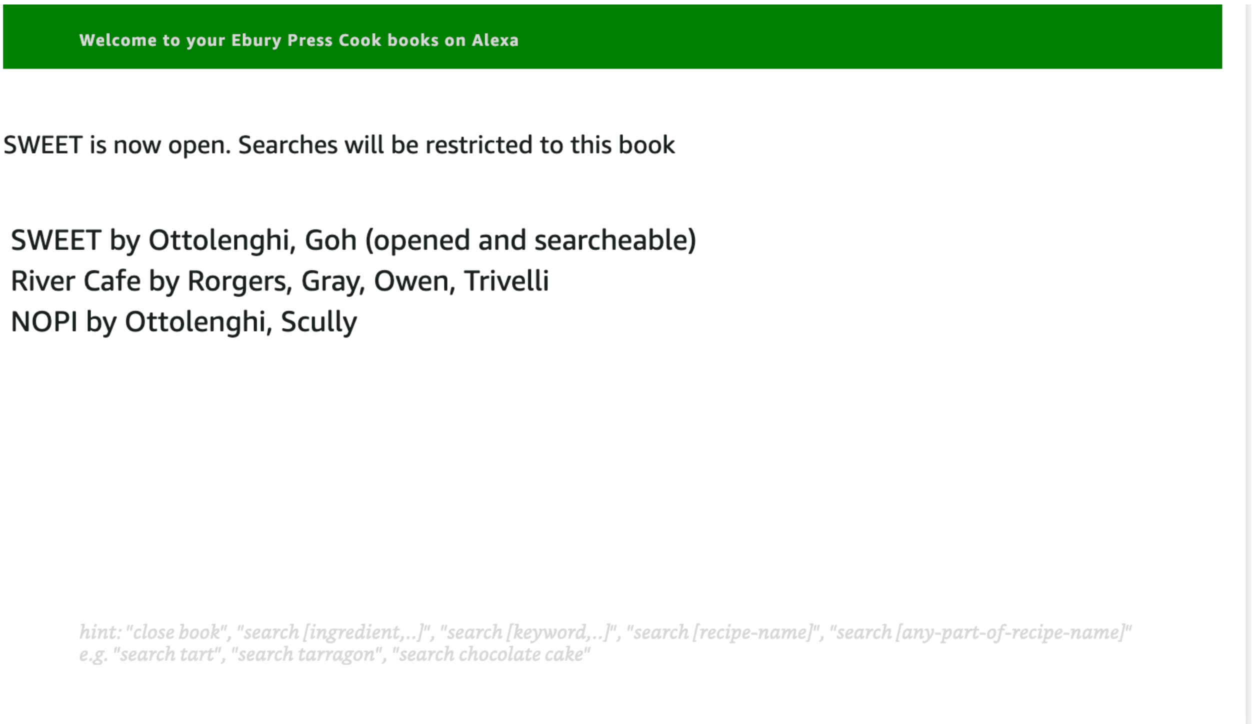
Listed below are the books registered to this device. Searches will be applied to all these books unless you open one

SWEET by Ottolenghi, Goh
River Cafe by Rogers, Gray, Owen, Trivelli
NOPI by Ottolenghi, Scully

hint: "open book River Cafe ", "search [ingredient,..]", "search [keyword,..]", "search [recipe-name]", "search [any-part-of-recipe-name]" e.g. "search tart", "search tarragon", "search chocolate cake"

*Cook Said: "**alexa <app launch phrase>**"*

The above shows the HOME screen when there are books registered to the email address. In this case there are three books. Note the hints at the bottom of the screen to help a cook unfamiliar with the App with what-to-ask next phrases. The hints are context sensitive.



Welcome to your Ebury Press Cook books on Alexa

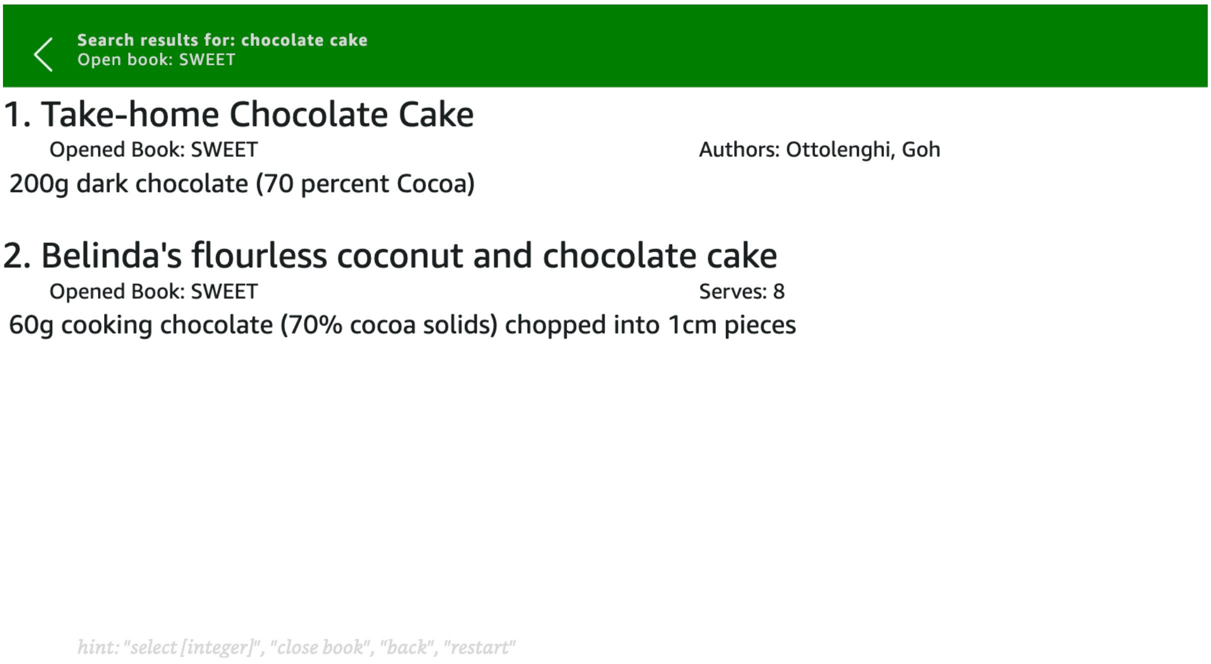
SWEET is now open. Searches will be restricted to this book

SWEET by Ottolenghi, Goh (opened and searcheable)
River Cafe by Rorgers, Gray, Owen, Trivelli
NOPI by Ottolenghi, Scully

hint: "close book", "search [ingredient,..]", "search [keyword,..]", "search [recipe-name]", "search [any-part-of-recipe-name]" e.g. "search tart", "search tarragon", "search chocolate cake"

Cook Said: "alexa open sweet"

The above screen is the result of the cook opening the book SWEET to restrict a search to recipes in that book.



Search results for: chocolate cake
Open book: SWEET

1. Take-home Chocolate Cake
Opened Book: SWEET Authors: Ottolenghi, Goh
200g dark chocolate (70 percent Cocoa)

2. Belinda's flourless coconut and chocolate cake
Opened Book: SWEET Serves: 8
60g cooking chocolate (70% cocoa solids) chopped into 1cm pieces

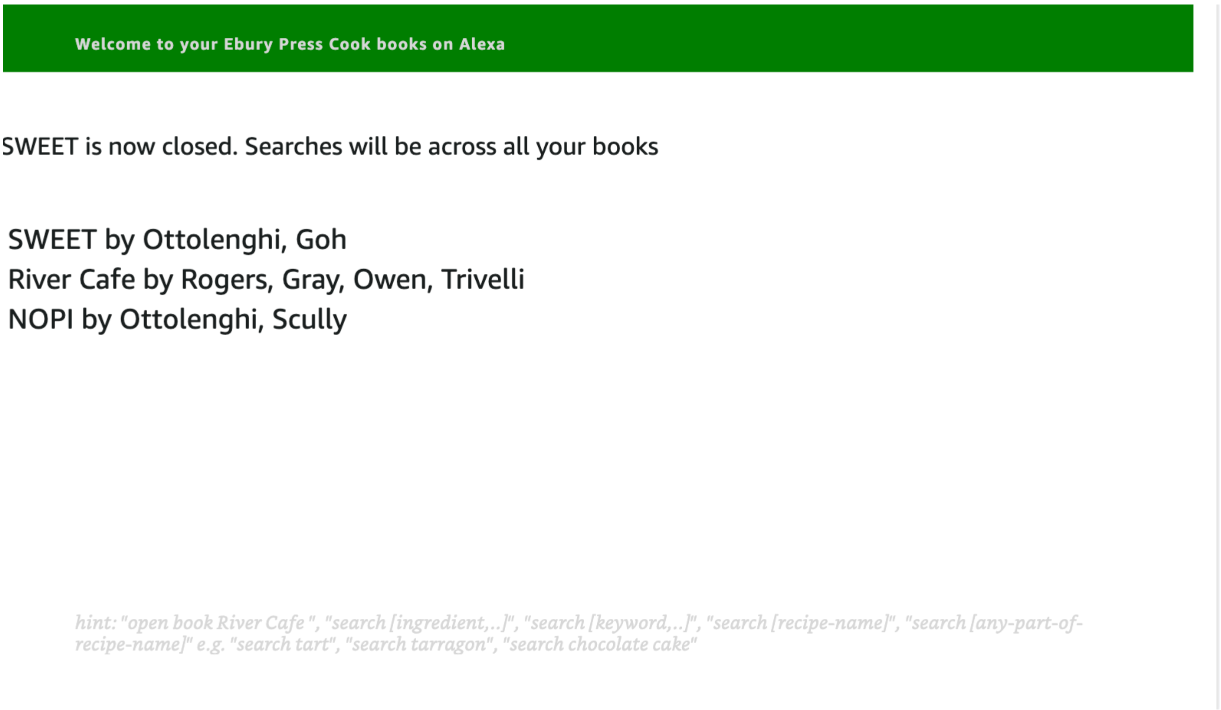
hint: "select [integer]", "close book", "back", "restart"

Cook Said: "alexa search chocolate cakes"

Alexa finds two recipes (both from the same "open" book) from the rather limited number of recipes in the development database. Note the App has identified "chocolate" as an ingredient in the search phrase and displayed its entry from the ingredient list of the associated recipe. This allows the cook to verify the quantity or variety of an ingredient at a glance, without having to open the recipe to determine how much chocolate is required in the recipe. Very handy when there are a lot of recipes in the list.

If there are two or more ingredients in the search phrase then the ingredient with the largest quantity is displayed. Recipe knowledge, like this examples shows, is accessible to the program as virtually everything about a recipe has been tokenised (see JSON schema in Appendix A). The App can make intelligent decisions about most aspects of a recipe as a result. Also, note that the header reminds the cook that the book SWEET is "open" and each search entry has "Opened Book: SWEET" as another reminder that the results are sourced from one book only.

From a Dynamobdb IO perspective, the above screen consumed 3 reads for a total of 1.5 read capacity units (rcu in Dynamo speak). In fact it would be 1.5 rcu's if tens of recipes were returned. The App has gone to considerably lengths to optimise IO.



Welcome to your Ebury Press Cook books on Alexa

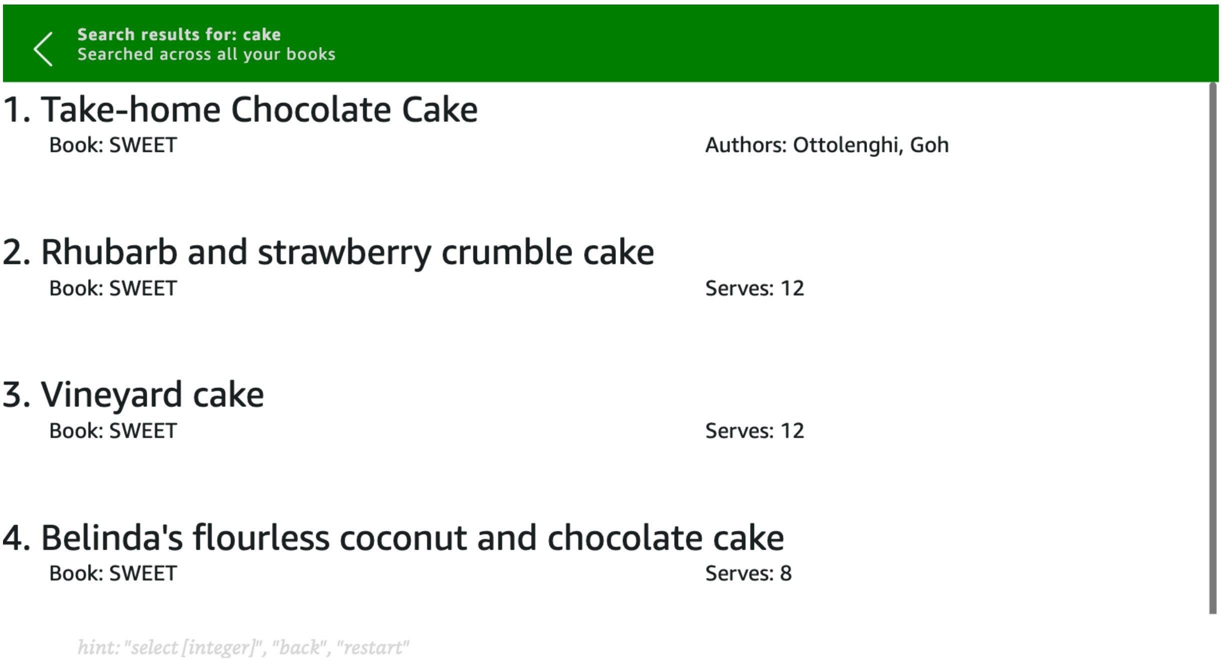
SWEET is now closed. Searches will be across all your books

SWEET by Ottolenghi, Goh
River Cafe by Rogers, Gray, Owen, Trivelli
NOPI by Ottolenghi, Scully

hint: "open book River Cafe ", "search [ingredient,..]", "search [keyword,..]", "search [recipe-name]", "search [any-part-of-recipe-name]" e.g. "search tart", "search tarragon", "search chocolate cake"

Cook Said: "alexa close book"

In this case the cook has decided he doesn't have enough chocolate to make the cake so has closed the book to allow the next search to scan all books. The App will automatically go back to the HOME screen when a book is closed.



Search results for: cake
Searched across all your books

1. Take-home Chocolate Cake
   Book: SWEET — Authors: Ottolenghi, Goh

2. Rhubarb and strawberry crumble cake
   Book: SWEET — Serves: 12

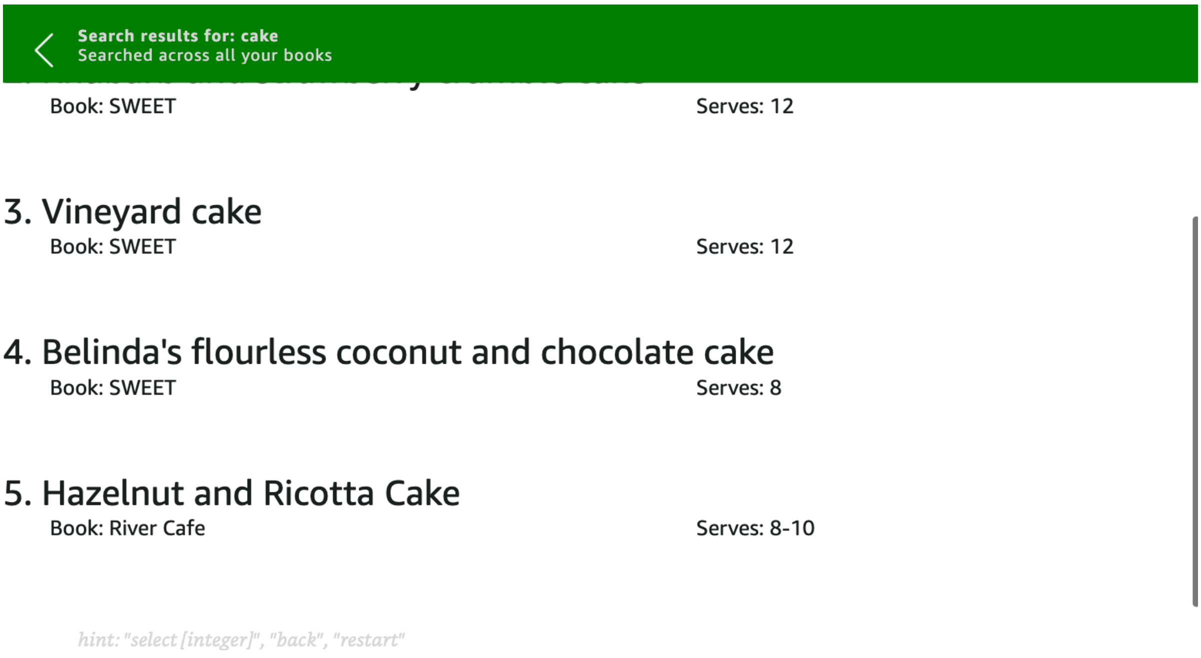
3. Vineyard cake
   Book: SWEET — Serves: 12

4. Belinda's flourless coconut and chocolate cake
   Book: SWEET — Serves: 8

hint: "select [integer]", "back", "restart"

Cook Said: "alexa search cake"

The cook decides to do a general query across all books for cakes. The App will return all recipes that have "cake" in the recipe name or has the keyword "cake" assigned to it. As no ingredient appears in the search phrase the output does not include an ingredient quantity.



Search results for: cake
Searched across all your books

Book: SWEET   Serves: 12

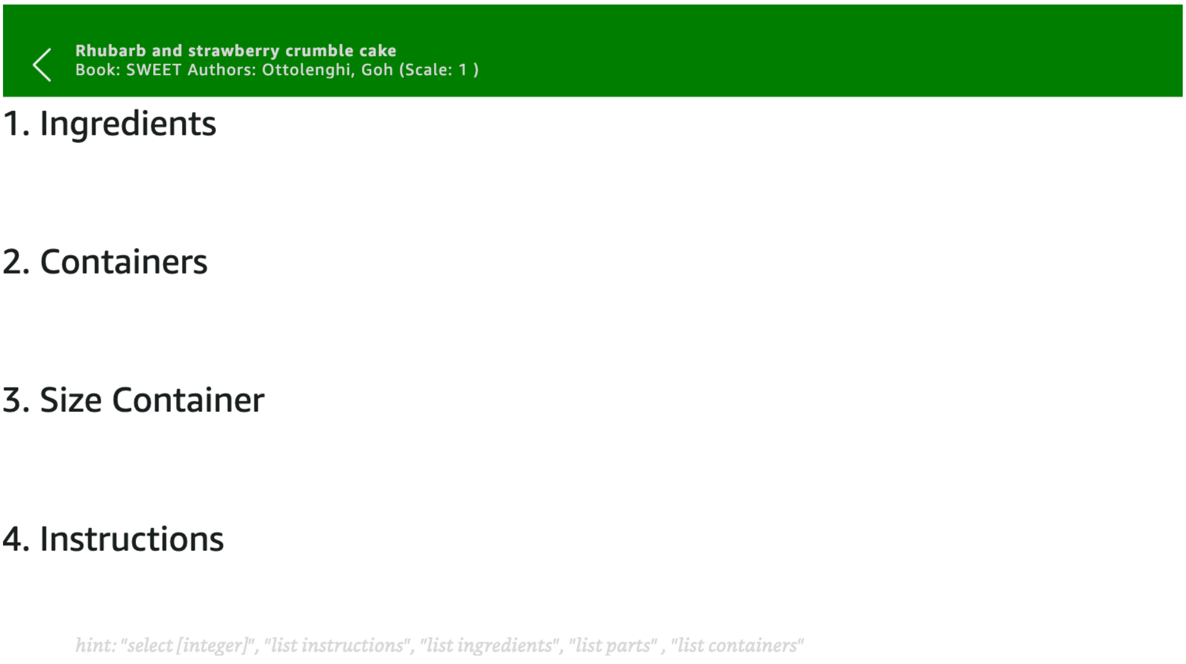
3. Vineyard cake
Book: SWEET   Serves: 12

4. Belinda's flourless coconut and chocolate cake
Book: SWEET   Serves: 8

5. Hazelnut and Ricotta Cake
Book: River Cafe   Serves: 8-10

hint: "select [integer]", "back", "restart"

*The cook scrolls the display up to reveal the last results. Note the last entry comes from a different book. The cook has decided on item 2 from the above list, **Rhubarb and Strawberry crumble cake**, which results in the next screen.*



Rhubarb and strawberry crumble cake
Book: SWEET Authors: Ottolenghi, Goh (Scale: 1)

1. Ingredients

2. Containers

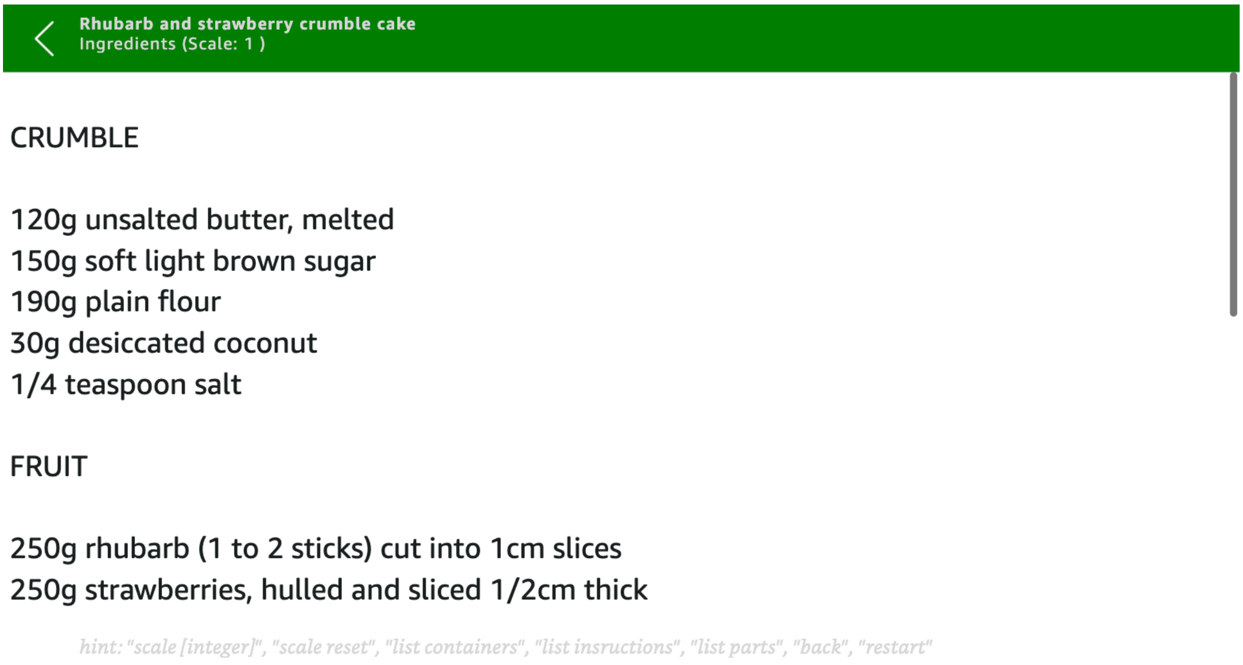
3. Size Container

4. Instructions

hint: "select [integer]", "list instructions", "list ingredients", "list parts" , "list containers"

Cook Said*: "alexa select two"*

This is the Recipe screen. Note the header displays the recipe name. There are either 3 or 4 options available. This recipe happens to allow the cook to override the container size so option 3 (Size Container) is available, otherwise it is not shown. Changing the container size results in scaling of the ingredient quantities.
*The cook is interested in **viewing the ingredients.** As shown in the hints, he has a few ways to do that. Use the verbal command "list ingredients", use the verbal command "select one" (item one in the above menu) or use the touch screen to select item 1.*



Rhubarb and strawberry crumble cake
Ingredients (Scale: 1)

CRUMBLE

120g unsalted butter, melted
150g soft light brown sugar
190g plain flour
30g desiccated coconut
1/4 teaspoon salt

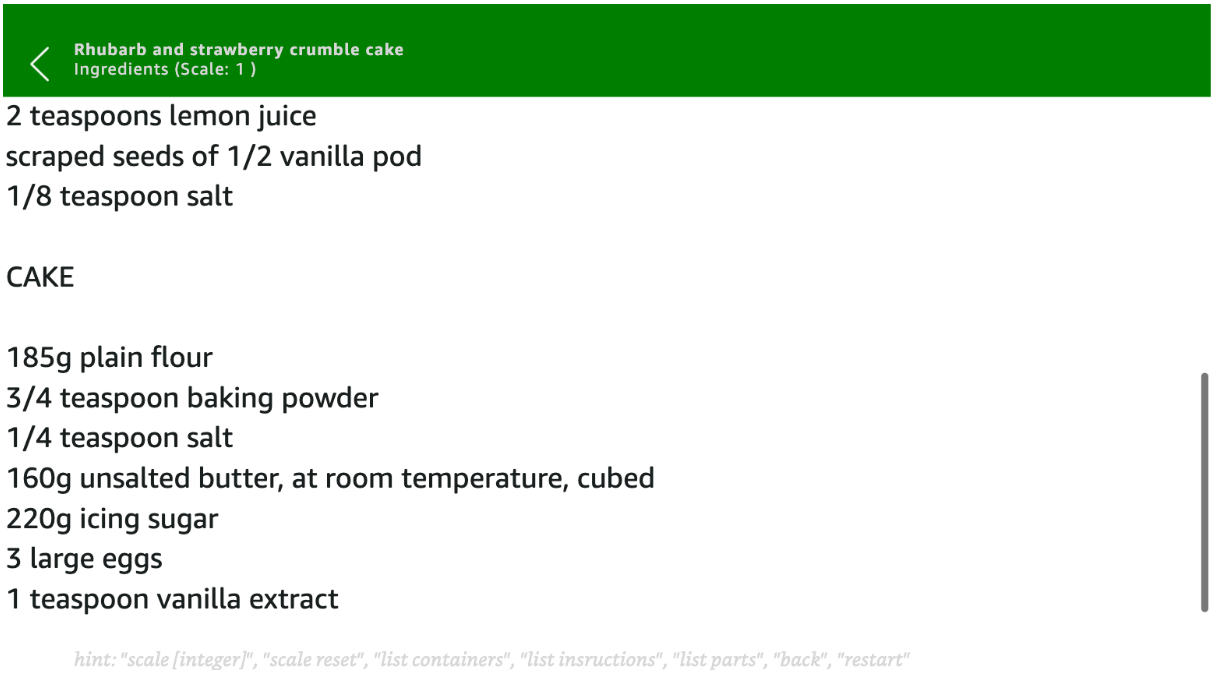
FRUIT

250g rhubarb (1 to 2 sticks) cut into 1cm slices
250g strawberries, hulled and sliced 1/2cm thick

hint: "scale [integer]", "scale reset", "list containers", "list insructions", "list parts", "back", "restart"

Cook Said: "alexa select one"

Above is the first page of the Ingredients screen. The ingredient listing will be very close to the format used in the associated book. Note the listing is divided into parts (Crumble, Fruit shown here), just as it is in SWEET. Parts are another attribute of a recipe.



Rhubarb and strawberry crumble cake
Ingredients (Scale: 1)

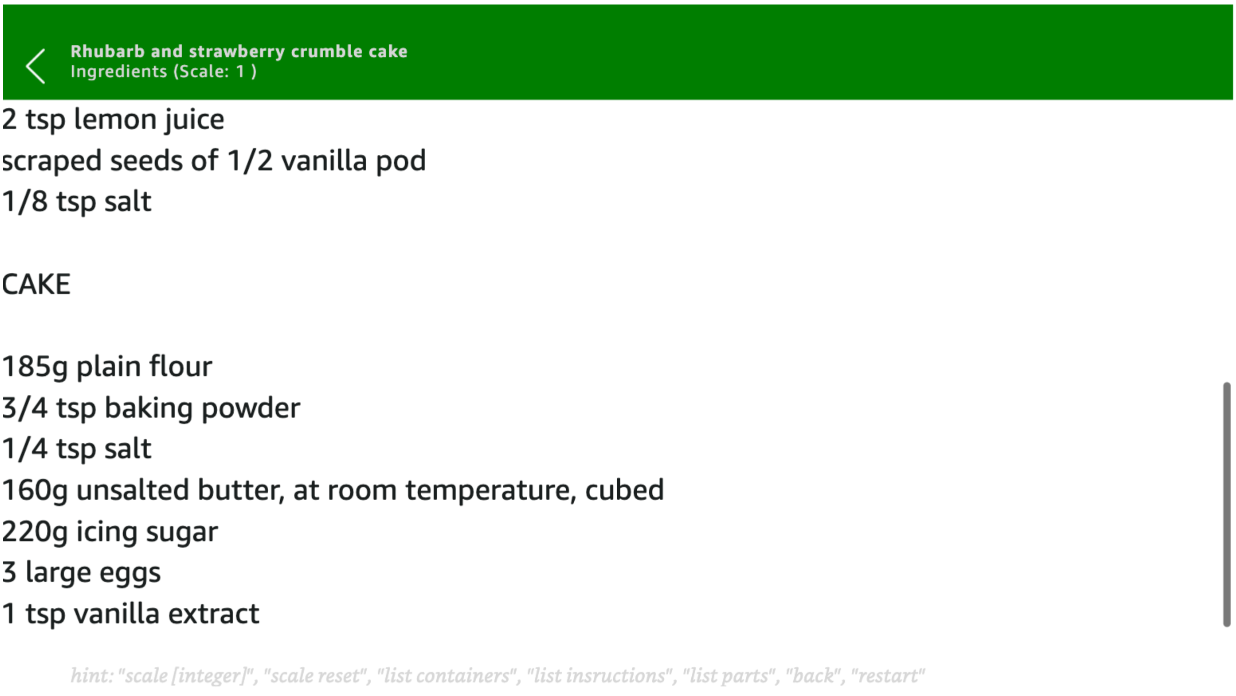
2 teaspoons lemon juice
scraped seeds of 1/2 vanilla pod
1/8 teaspoon salt

CAKE

185g plain flour
3/4 teaspoon baking powder
1/4 teaspoon salt
160g unsalted butter, at room temperature, cubed
220g icing sugar
3 large eggs
1 teaspoon vanilla extract

hint: "scale [integer]", "scale reset", "list containers", "list insructions", "list parts", "back", "restart"

*The cook scrolls the display up to reveal another "part". Note the screen header lists the recipe name and screen name as well as the current scale used for the ingredients. In this case it is 1, meaning the quantities have not been adjusted from the original recipe. Also note the unit "teaspoon" is shown not abbreviated, which is unusual. This is software configurable. The next screen shows it abbreviated after the software was reconfigured. The **session state** is maintained for a few days, meaning if the cook exists the App now and relaunches it within three days the App will start from where it left off, namely the Ingredient screen.*



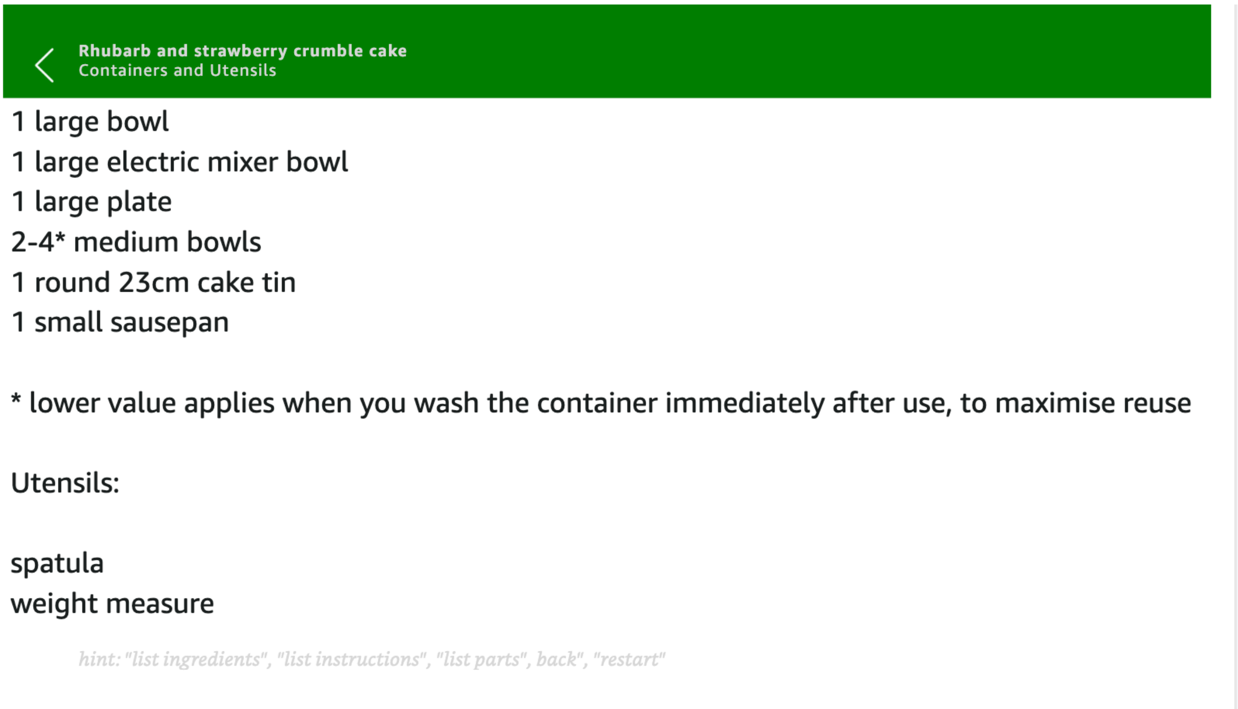
2 tsp lemon juice
scraped seeds of 1/2 vanilla pod
1/8 tsp salt

CAKE

185g plain flour
3/4 tsp baking powder
1/4 tsp salt
160g unsalted butter, at room temperature, cubed
220g icing sugar
3 large eggs
1 tsp vanilla extract

hint: "scale [integer]", "scale reset", "list containers", "list insructions", "list parts", "back", "restart"

Teaspoon is now abbreviated. While this is only software configurable at present, future versions could expose this feature directly to the cook via new verbal commands, like "unit teaspoon short" or "unit teaspoon long". Similarly for "g", "unit gram long", to display grams as unabbreviated. To adjust all units to their long or short form say "unit long" or "unit short". The configuration would be persisted to the database forever using the device id as the key, beyond the current session data of a few days.



1 large bowl
1 large electric mixer bowl
1 large plate
2-4* medium bowls
1 round 23cm cake tin
1 small sausepan

* lower value applies when you wash the container immediately after use, to maximise reuse

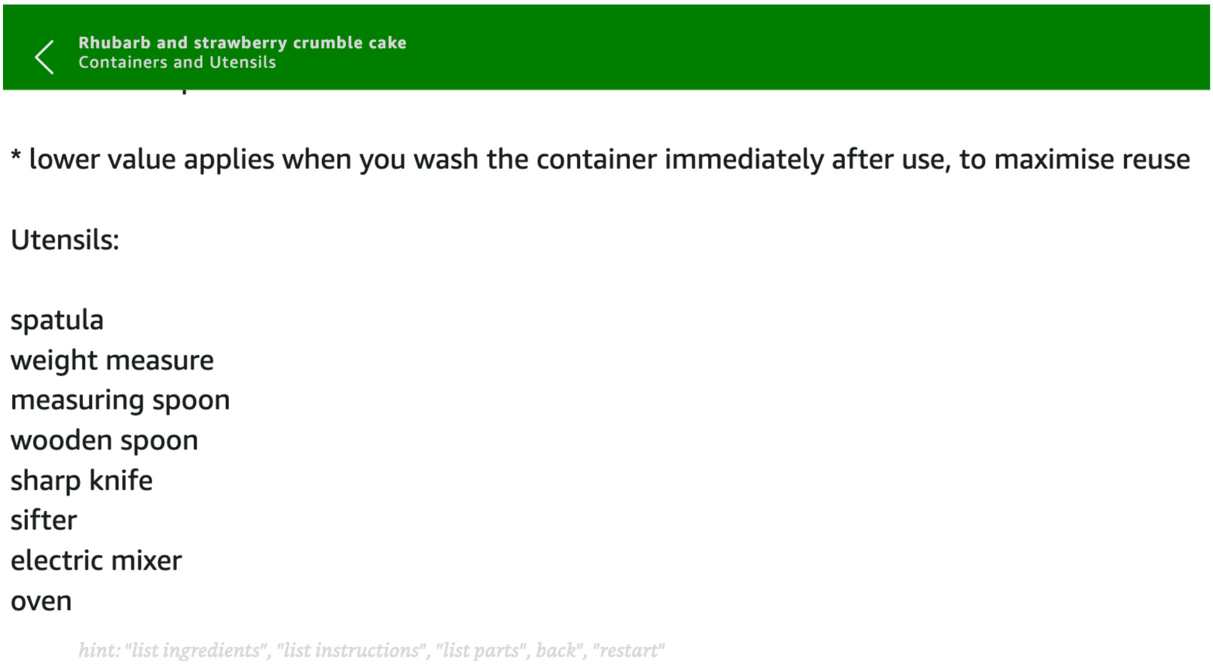
Utensils:

spatula
weight measure

hint: "list ingredients", "list instructions", "list parts", back", "restart"

Cook Said: "alexa list containers"

The cook now decides to check out what containers are required. Note they can go directly to the container screen from the ingredient screen, using "list containers". Alternatively, they could use the back button (in the header) to go back to the Recipe menu and select item 2. Obviously the verbal command is more convenient and faster.



Rhubarb and strawberry crumble cake
Containers and Utensils

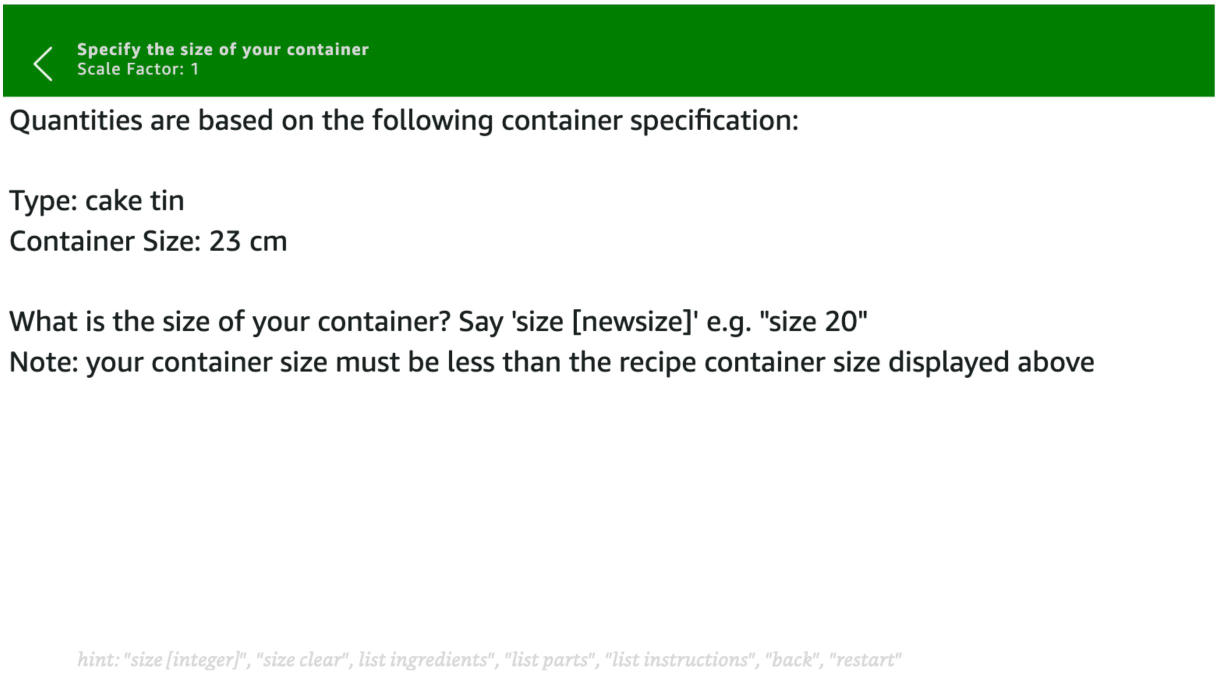
* lower value applies when you wash the container immediately after use, to maximise reuse

Utensils:

spatula
weight measure
measuring spoon
wooden spoon
sharp knife
sifter
electric mixer
oven

hint: "list ingredients", "list instructions", "list parts", back", "restart"

The cook scrolls the screen up to reveal the remaining utensils. Containers and utensils are known to the App as they have both been tokenised and are available to the App in anyway you can think of.
The cook observes from the first container screen that the recipe demands a 23cm round cake tin. Unfortunately the cook only has a 20cm cake tin at hand, so they decide to use the ***Size Container*** *feature to adjust the recipe quantities to suit their container.*



Specify the size of your container
Scale Factor: 1

Quantities are based on the following container specification:

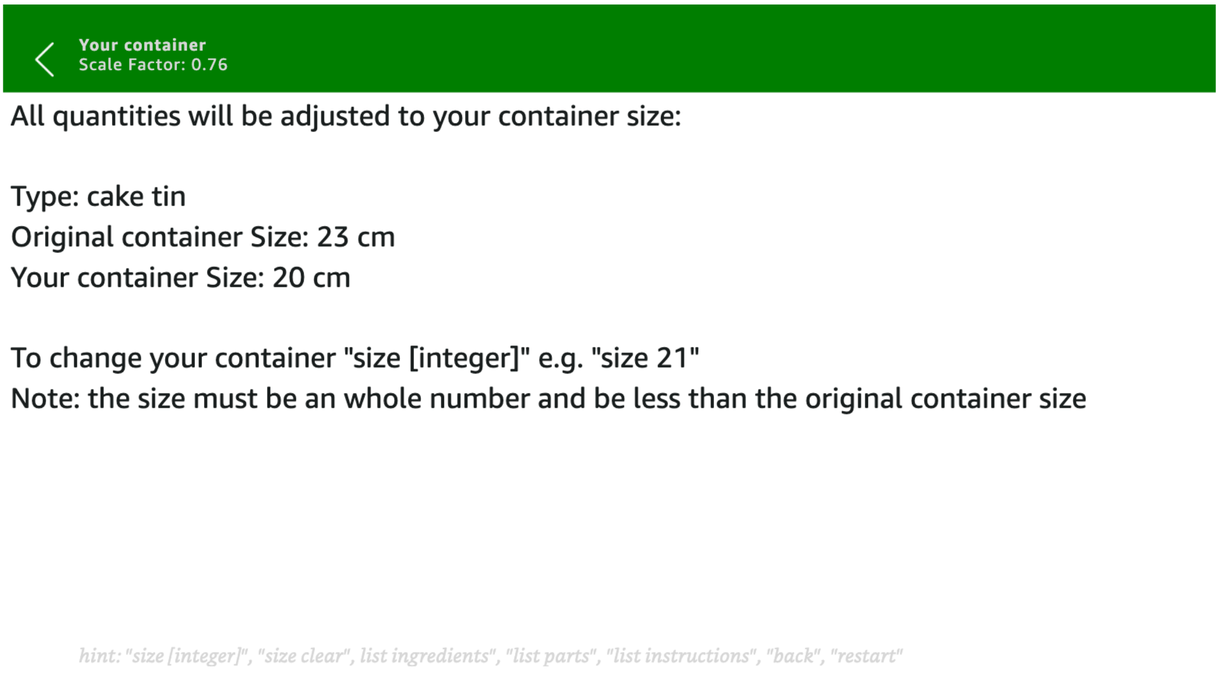
Type: cake tin
Container Size: 23 cm

What is the size of your container? Say 'size [newsize]' e.g. "size 20"
Note: your container size must be less than the recipe container size displayed above

hint: "size [integer]", "size clear", list ingredients", "list parts", "list instructions", "back", "restart"

Cook Said: "Alexa back"
"Alexa select three"

The cook uses the above verbal commands to navigate the menu hierarchy to the "Size Container" screen shown above. It confirms the recipe requires a 23 cm cake tin.



Your container
Scale Factor: 0.76

All quantities will be adjusted to your container size:

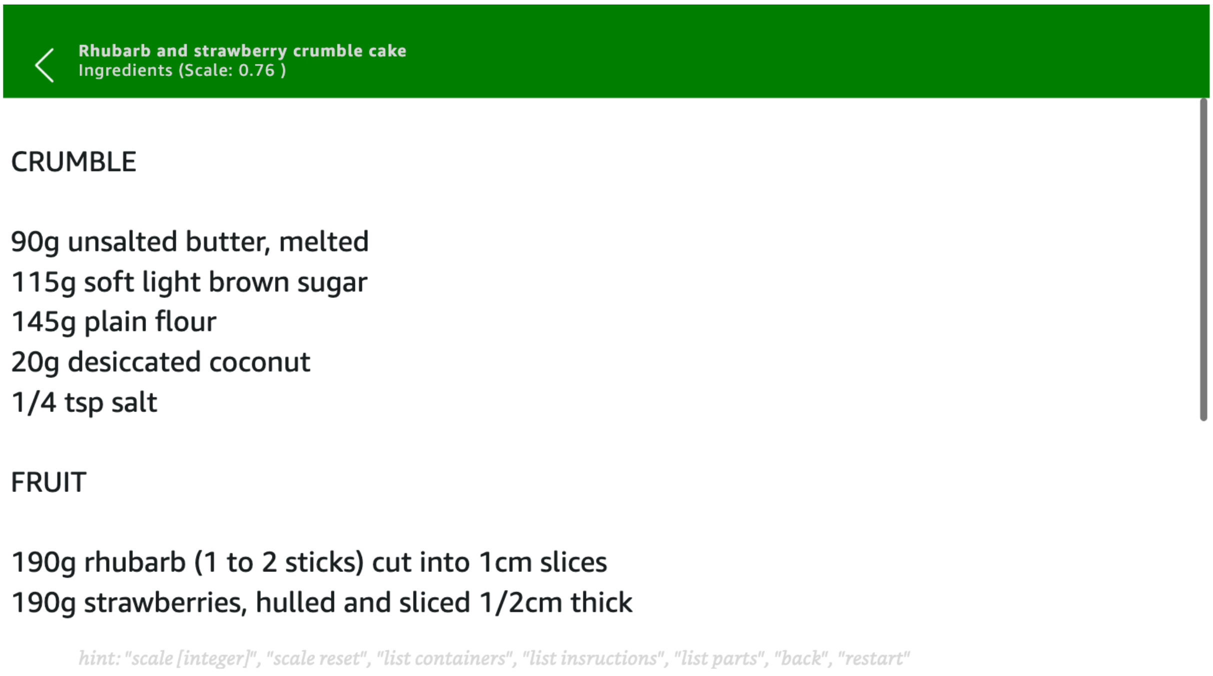
Type: cake tin
Original container Size: 23 cm
Your container Size: 20 cm

To change your container "size [integer]" e.g. "size 21"
Note: the size must be an whole number and be less than the original container size

hint: "size [integer]", "size clear", list ingredients", "list parts", "list instructions", "back", "restart"

Cook Said: "alexa size twenty"

The cook stipulates the size of his cake tin. The App will apply a multiplying factor of 0.76 to all ingredients, as indicated in the header. The cook now wants to view the ingredient listing again given the change in requirements.



Rhubarb and strawberry crumble cake
Ingredients (Scale: 0.76)

CRUMBLE

90g unsalted butter, melted
115g soft light brown sugar
145g plain flour
20g desiccated coconut
1/4 tsp salt

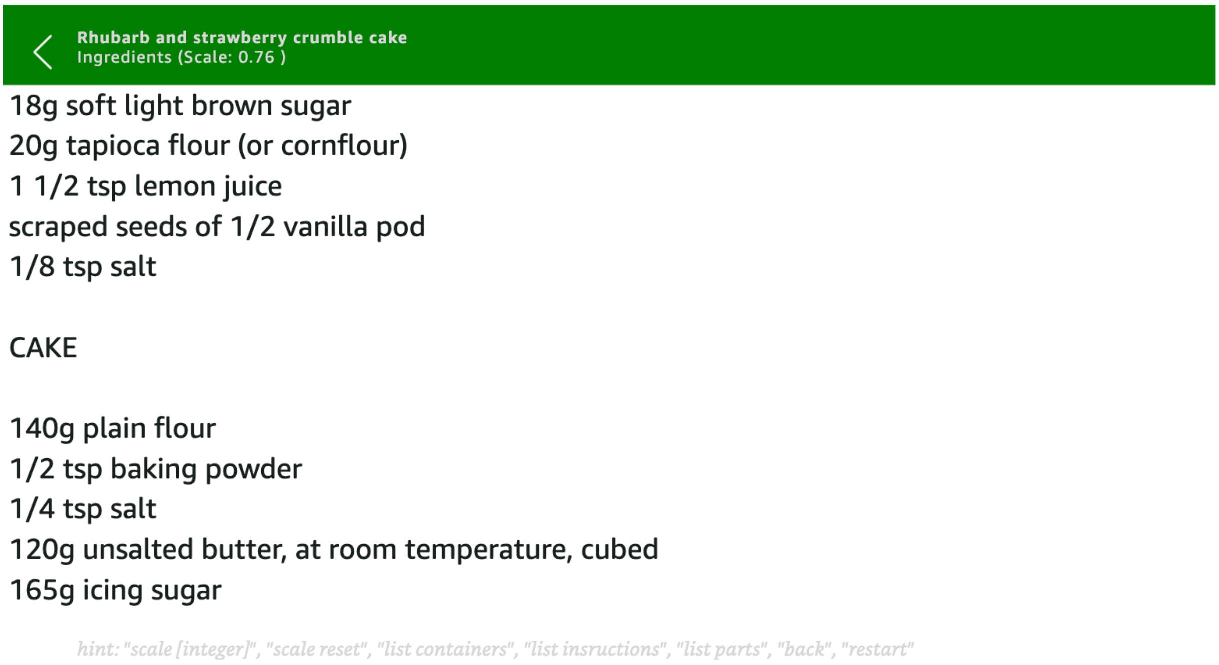
FRUIT

190g rhubarb (1 to 2 sticks) cut into 1cm slices
190g strawberries, hulled and sliced 1/2cm thick

hint: "scale [integer]", "scale reset", "list containers", "list insructions", "list parts", "back", "restart"

Cook Said: "alexa list ingredients"

Note all ingredients have been adjusted and scale factor of 0.76 in the header as a reminder. The App will round any quantity over 20 to the nearest 0 or 5. Teaspoons are calibrated from 1 to 3/4 to 1/2 to 1/4 to 1/8 to "pinch of".



Rhubarb and strawberry crumble cake
Ingredients (Scale: 0.76)

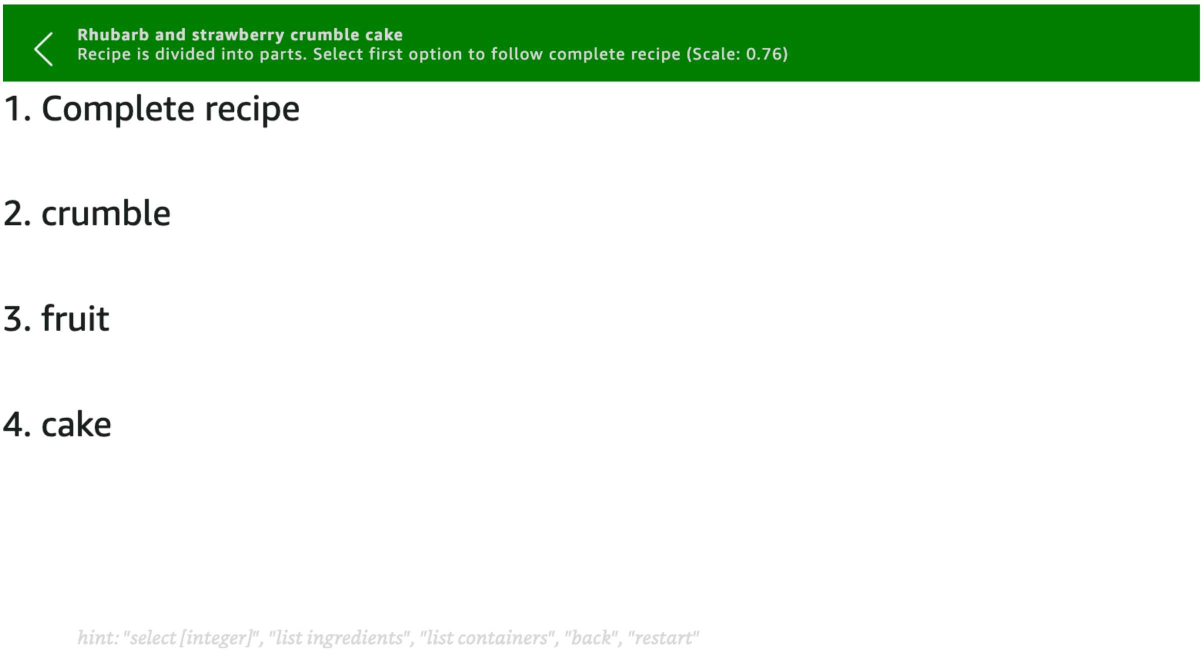
18g soft light brown sugar
20g tapioca flour (or cornflour)
1 1/2 tsp lemon juice
scraped seeds of 1/2 vanilla pod
1/8 tsp salt

CAKE

140g plain flour
1/2 tsp baking powder
1/4 tsp salt
120g unsalted butter, at room temperature, cubed
165g icing sugar

hint: "scale [integer]", "scale reset", "list containers", "list insructions", "list parts", "back", "restart"

The cook scrolls the display up to show more of the ingredients. Note the lemon juice quantity is defined as 1 1/2 not 1.5. This is configurable. At this stage the cook is prepared to start cooking..



Rhubarb and strawberry crumble cake
Recipe is divided into parts. Select first option to follow complete recipe (Scale: 0.76)

1. Complete recipe

2. crumble

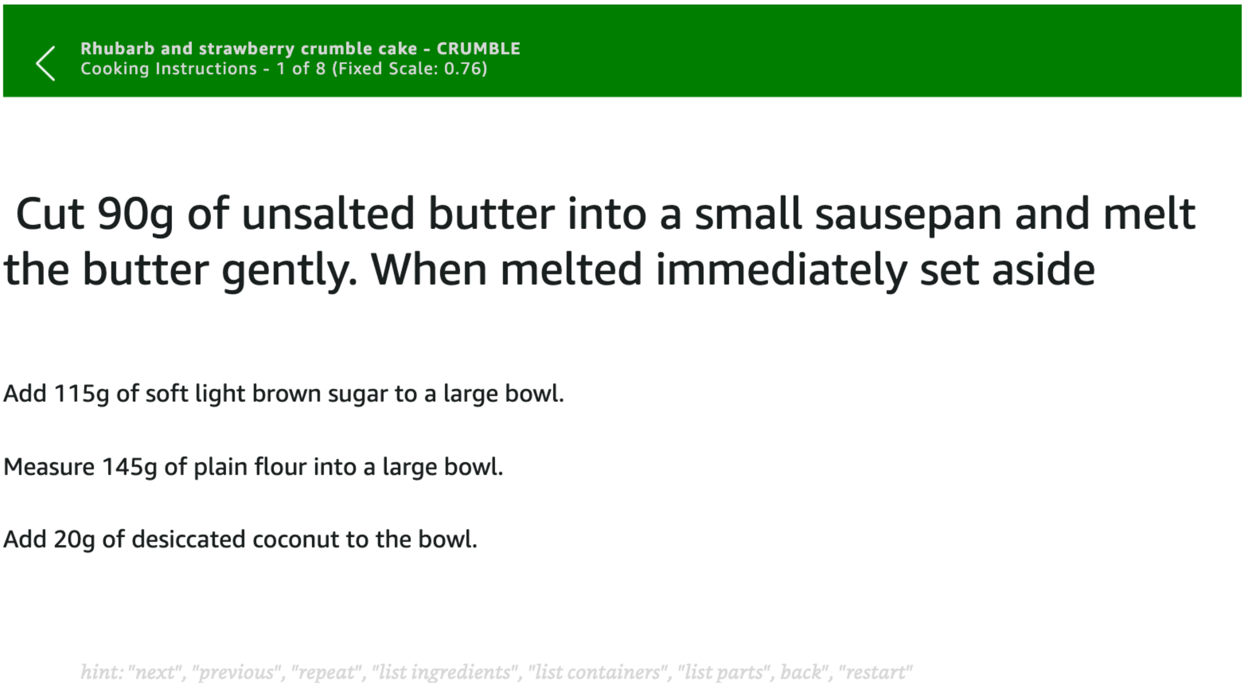
3. fruit

4. cake

hint: "select [integer]", "list ingredients", "list containers", "back", "restart"

Cook Said: "alexa list instructions"

As the recipe has been configured into "parts" and there is no active part yet, the App first lists the Part Menu before proceeding to cooking instructions. The cook can select the desired "part" or select item 1 for the complete recipe. Today, the cook is interested to learn what is involved in making the crumble, so selects item 2.



Rhubarb and strawberry crumble cake - CRUMBLE
Cooking Instructions - 1 of 8 (Fixed Scale: 0.76)

Cut 90g of unsalted butter into a small sausepan and melt the butter gently. When melted immediately set aside

Add 115g of soft light brown sugar to a large bowl.

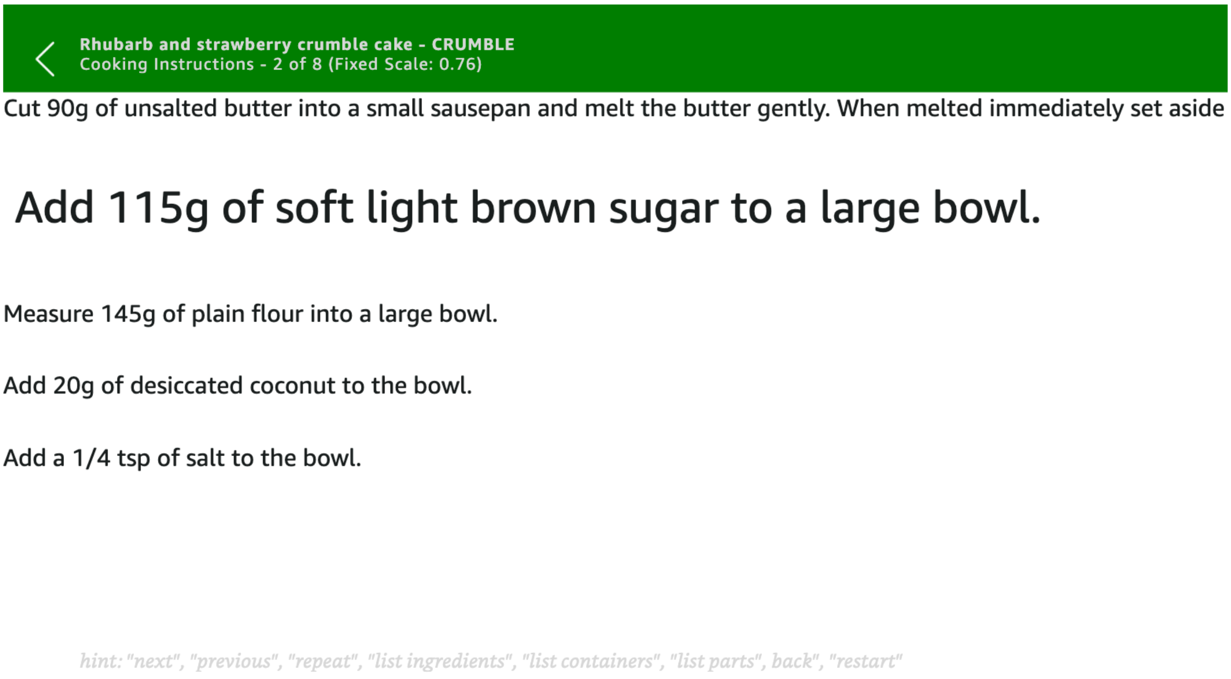
Measure 145g of plain flour into a large bowl.

Add 20g of desiccated coconut to the bowl.

hint: "next", "previous", "repeat", "list ingredients", "list containers", "list parts", back", "restart"

Cook Said: "alexa select two"

*The cook selects item 2 from the Part Menu which makes Crumble the active recipe part. The above **cooking instruction** screen is then displayed. The cook rarely needs to view this screen as Alexa will automatically recite the current instruction (identified in large font) as soon as the screen is displayed and thereafter Alexa will respond to verbal commands to recite the remainder. Even so, the header displays the recipe name and "recipe part" name along with the number of the current instruction and total number of instructions for the part. Note that all the quantities reflect their adjusted values as a result of scaling by 0.76.*
Now to go to the next instruction…



Rhubarb and strawberry crumble cake - CRUMBLE
Cooking Instructions - 2 of 8 (Fixed Scale: 0.76)

Cut 90g of unsalted butter into a small sausepan and melt the butter gently. When melted immediately set aside

Add 115g of soft light brown sugar to a large bowl.

Measure 145g of plain flour into a large bowl.

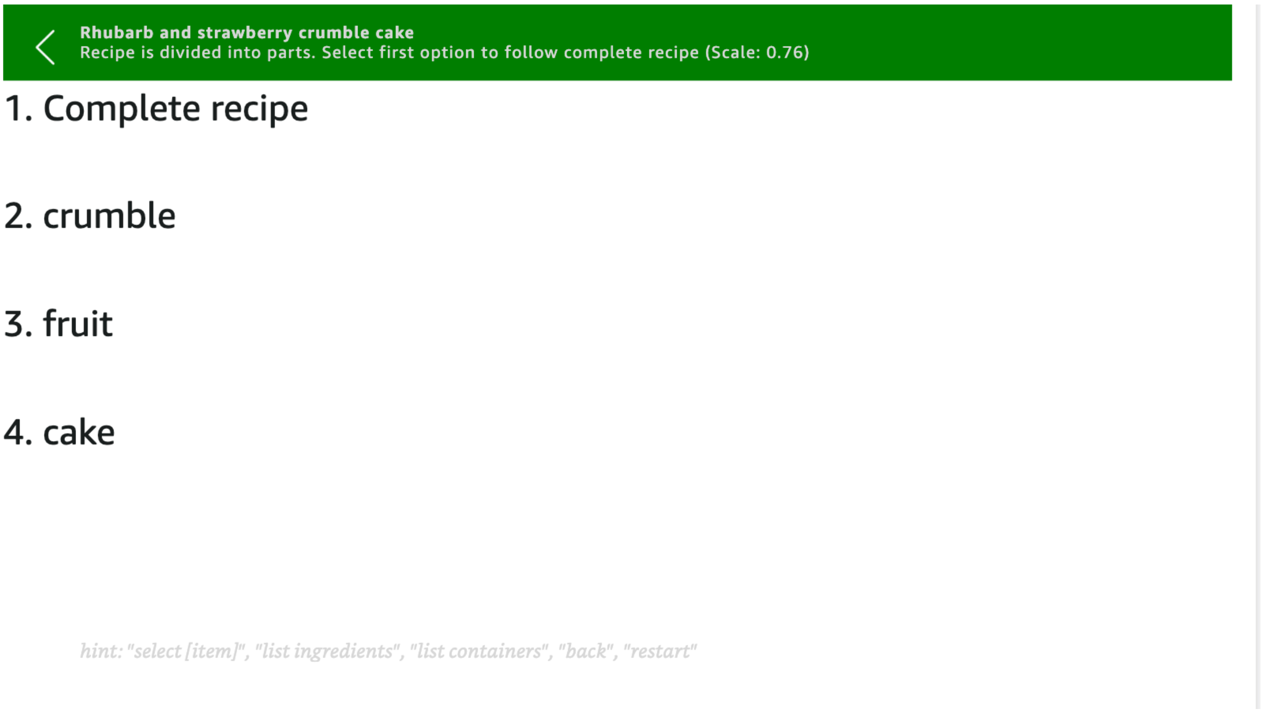
Add 20g of desiccated coconut to the bowl.

Add a 1/4 tsp of salt to the bowl.

hint: "next", "previous", "repeat", "list ingredients", "list containers", "list parts", back", "restart"

Cook Said: "Alexa next"

Note the instruction counter has progressed to 2 and the previous instruction is now visible at the top of the screen. In development I have chosen to display only one of the previous instructions however, the whole screen design is up for review.
*The cook can make use of the following verbal commands mentioned in the hint at the bottom of the screen. "repeat' (or "say again") to get Alexa to recite the current instruction again, "previous" to go back to the previous screen at which point Alexa will recite the then current instruction (or previous in the above). Not shown in the hint is the "go to" command, which lets the cook go directly to the instruction specified. The cook can also go directly to the ingredients or container listings by saying "**list ingredients**" or "**list containers**", review the content and come back to the instruction screen by saying "**list instructions**" again. The App will maintain the current context, meaning it keeps track of the current instruction number and current "part" as the user navigates around the App.*
The cook has reviewed the crumble and now wants to see other parts of the recipe. He has the option to use the back arrow (in the header) to go back one in the menu hierarchy (as the Part Menu was the previous screen) or use the verbal command "list parts"



Rhubarb and strawberry crumble cake
Recipe is divided into parts. Select first option to follow complete recipe (Scale: 0.76)

1. Complete recipe

2. crumble

3. fruit

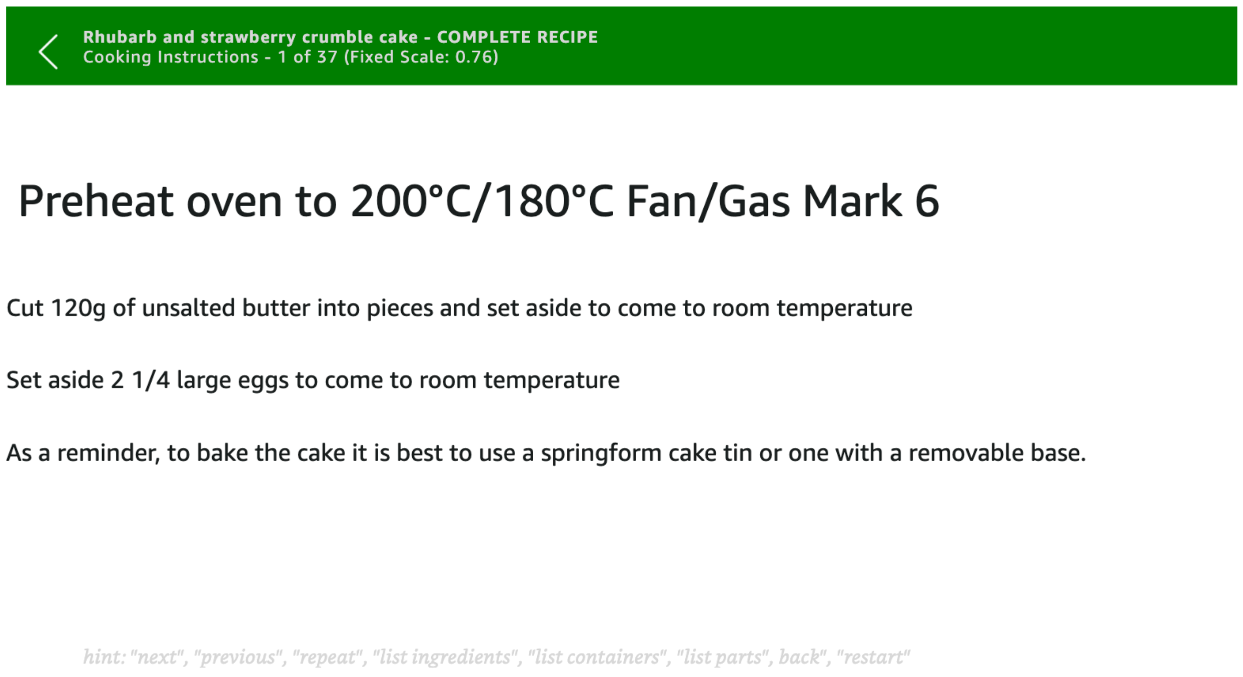
4. cake

hint: "select [item]", "list ingredients", "list containers", "back", "restart"

Cook Said: "Alexa list parts"

Back to the parts listing using the command "list parts", this time. The cook can say "list parts" from any screen once there is an active recipe.

While we are here, recipe parts can be aggregated by ingredients, as above, or by time, into instructions that can be carried out "before the day" and those that are performed "on the day", as an example. A recipe can be configured with both types of aggregation. Breaking up a recipe into parts is either specified by the chef or the recipe designer. Incorporating it as a feature of the App, relieves the cook from having to resolve this information from the recipe and goes some way to reducing the preparation time.



Rhubarb and strawberry crumble cake - COMPLETE RECIPE
Cooking Instructions - 1 of 37 (Fixed Scale: 0.76)

Preheat oven to 200°C/180°C Fan/Gas Mark 6

Cut 120g of unsalted butter into pieces and set aside to come to room temperature

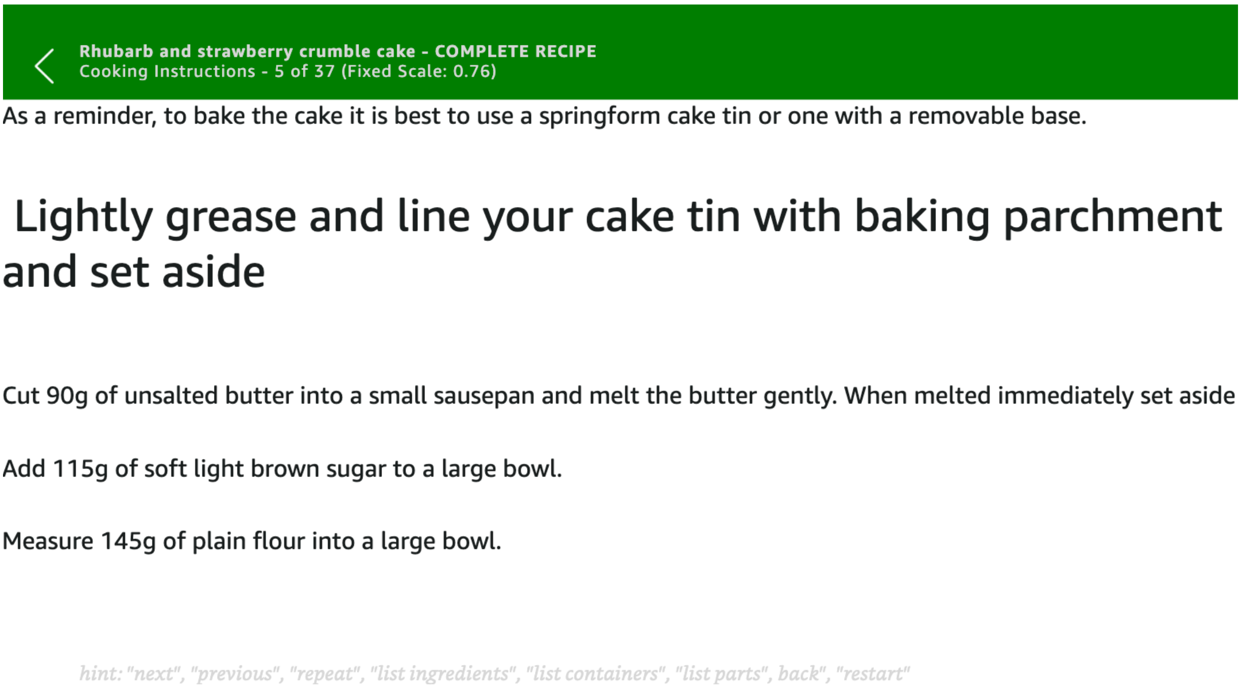
Set aside 2 1/4 large eggs to come to room temperature

As a reminder, to bake the cake it is best to use a springform cake tin or one with a removable base.

hint: "next", "previous", "repeat", "list ingredients", "list containers", "list parts", back", "restart"

Cook Said: "Alexa select one"

Back to the instruction screen. Note there are now 37 instructions as the cook selected the complete recipe. Typically the instructions are order into "prep" tasks, "measuring" tasks, and actual "cooking" tasks, although that is at the discretion of the recipe designer.
Alexa will recite the above instruction not as written but slightly different:
*"**preheat oven to 200 degrees celsius or 180 fan forced or gas mark 6**". Removing the second reference to celsius makes her sound almost human.*
Alexa will also recite all units in their non-abbreviated form even if it is displayed abbreviated.
Note the rather odd measure of 2 1/4 eggs. It is a result of scaling the original recipe by 0.76. While it is possible to measure fractions of an egg easily enough this is a reminder that there are often compromises that must be accommodated when scaling recipes.



Rhubarb and strawberry crumble cake - COMPLETE RECIPE
Cooking Instructions - 5 of 37 (Fixed Scale: 0.76)

As a reminder, to bake the cake it is best to use a springform cake tin or one with a removable base.

Lightly grease and line your cake tin with baking parchment and set aside

Cut 90g of unsalted butter into a small sausepan and melt the butter gently. When melted immediately set aside

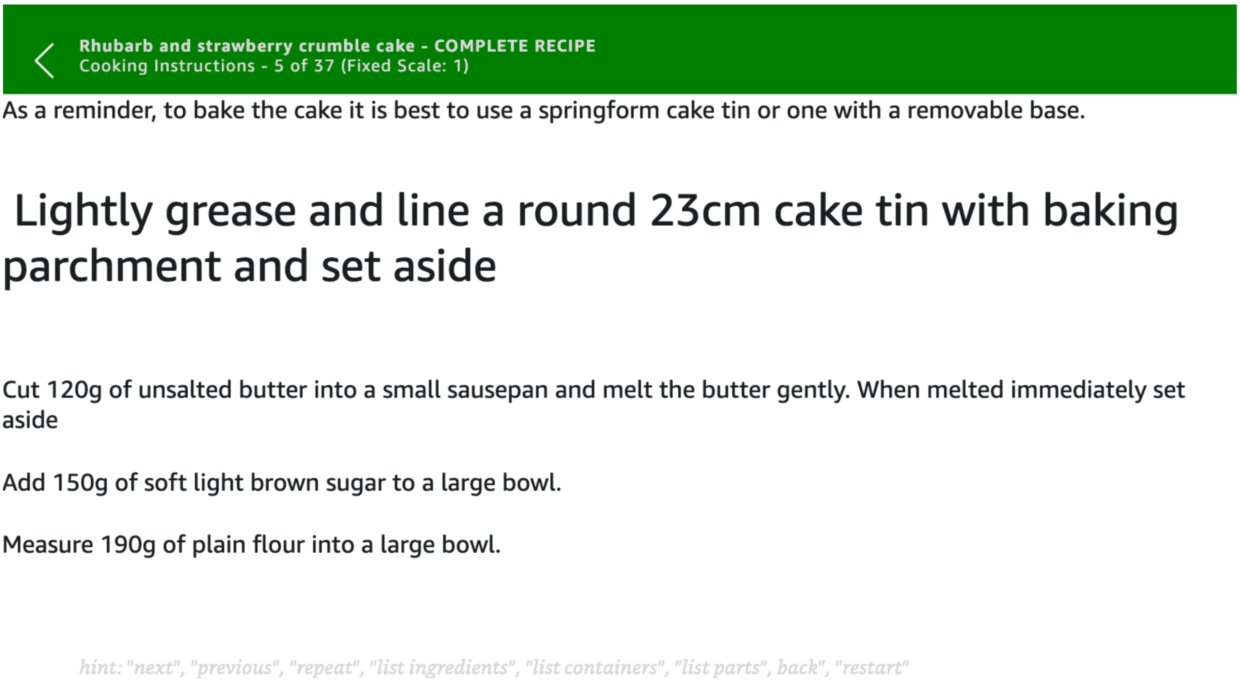
Add 115g of soft light brown sugar to a large bowl.

Measure 145g of plain flour into a large bowl.

hint: "next", "previous", "repeat", "list ingredients", "list containers", "list parts", back", "restart"

Cook Said: “Alexa go to five”

As another example of instruction personalisation the cook steps directly to instruction 5. Here it references “your cake tin” as the Alexa knows the cook has overridden the recommend container size with their own.
Screen shot below has the non-personalised instruction.



Rhubarb and strawberry crumble cake - COMPLETE RECIPE
Cooking Instructions - 5 of 37 (Fixed Scale: 1)

As a reminder, to bake the cake it is best to use a springform cake tin or one with a removable base.

Lightly grease and line a round 23cm cake tin with baking parchment and set aside

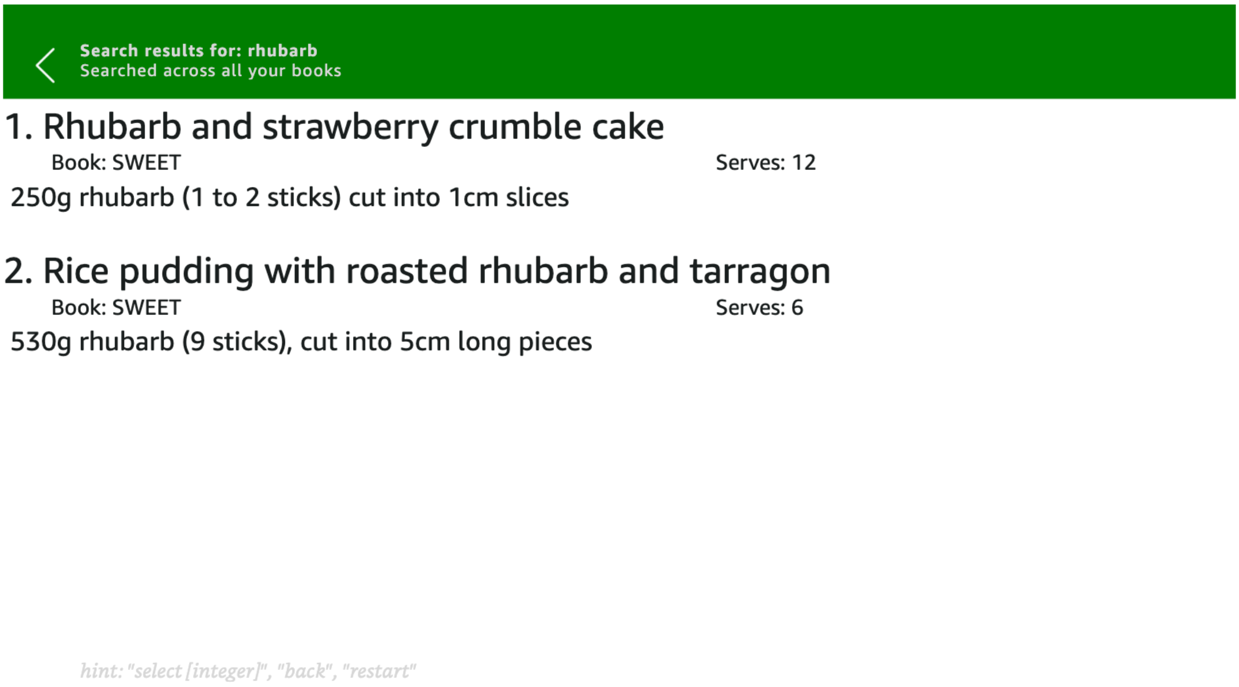
Cut 120g of unsalted butter into a small sausepan and melt the butter gently. When melted immediately set aside

Add 150g of soft light brown sugar to a large bowl.

Measure 190g of plain flour into a large bowl.

hint: "next", "previous", "repeat", "list ingredients", "list containers", "list parts", back", "restart"

The scale is 1 in this screen shot. This means the cook has not overridden the recipe’s container size, consequently instruction five references the recommended container size. Finally, to demonstrate multithreaded recipes we need to go back to the start and find a recipe that has a multithreaded component.



Search results for: rhubarb
Searched across all your books

1. Rhubarb and strawberry crumble cake

Book: SWEET Serves: 12

250g rhubarb (1 to 2 sticks) cut into 1cm slices

2. Rice pudding with roasted rhubarb and tarragon

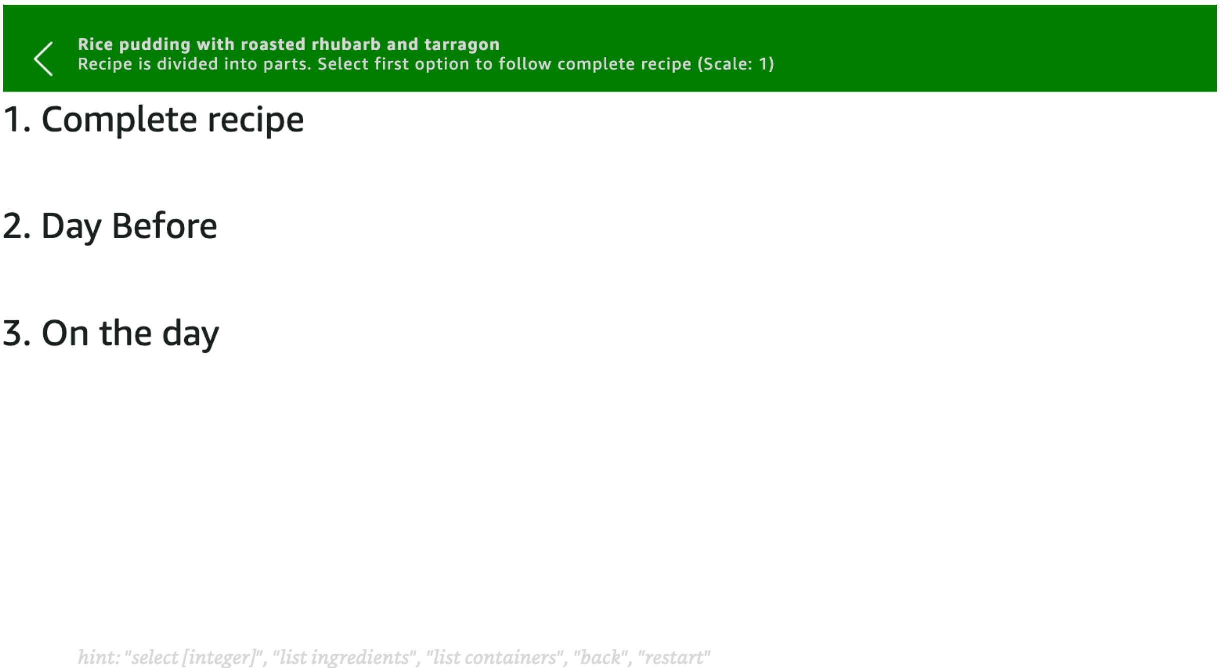
Book: SWEET Serves: 6

530g rhubarb (9 sticks), cut into 5cm long pieces

hint: "select [integer]", "back", "restart"

Cook Said: "Alexa restart"
"Alexa search rhubarb"

Restart goes directly to the Home screen and the search results for rhubarb are displayed above. The second recipe has a multi-threaded component so we will select that one.



Rice pudding with roasted rhubarb and tarragon
Recipe is divided into parts. Select first option to follow complete recipe (Scale: 1)

1. Complete recipe

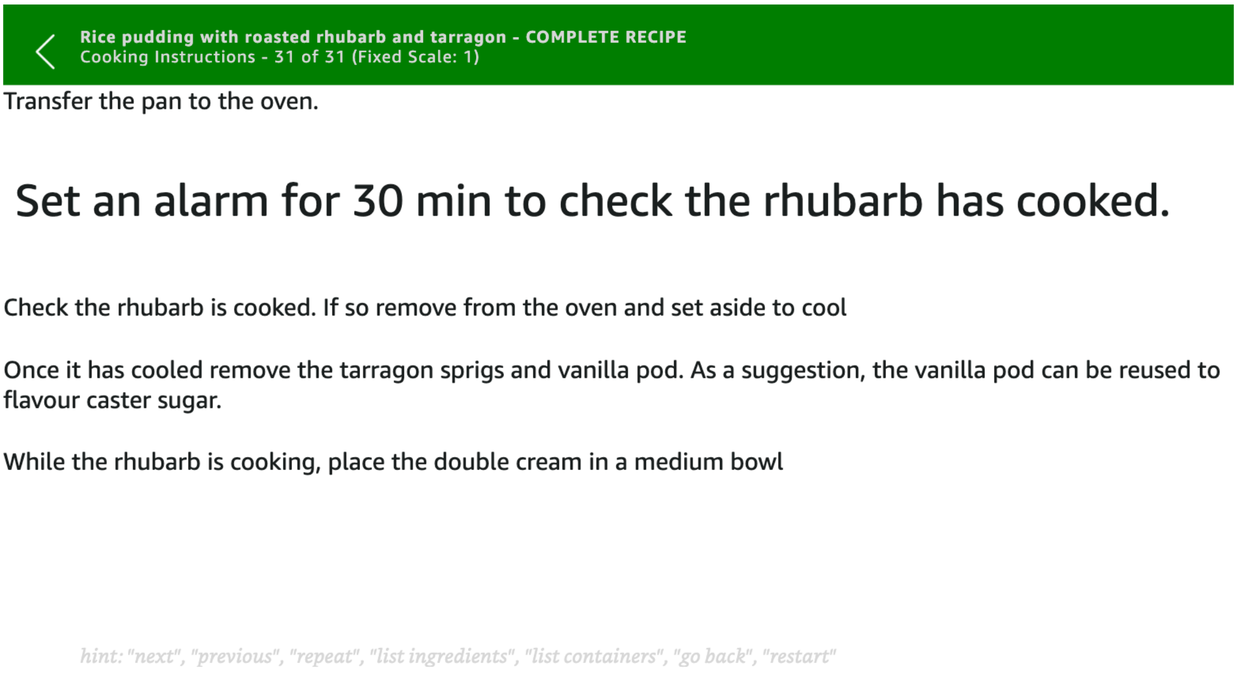
2. Day Before

3. On the day

hint: "select [integer]", "list ingredients", "list containers", "back", "restart"

Cook Said: "Alexa select two"
"Alexa list instructions"

This is the parts screen for the rice pudding recipe. Now he selects the complete recipe and goes to instruction 31. Note this recipe has also been divided into parts based on time



Rice pudding with roasted rhubarb and tarragon - COMPLETE RECIPE
Cooking Instructions - 31 of 31 (Fixed Scale: 1)

Transfer the pan to the oven.

Set an alarm for 30 min to check the rhubarb has cooked.

Check the rhubarb is cooked. If so remove from the oven and set aside to cool

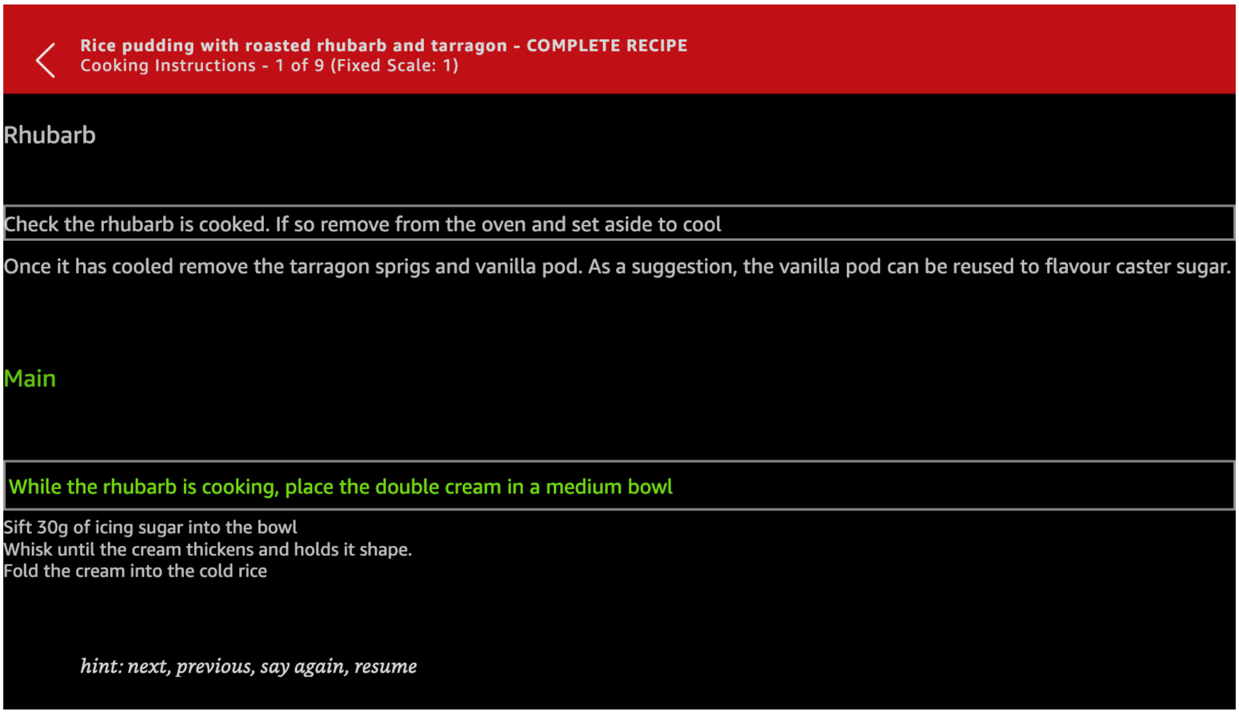
Once it has cooled remove the tarragon sprigs and vanilla pod. As a suggestion, the vanilla pod can be reused to flavour caster sugar.

While the rhubarb is cooking, place the double cream in a medium bowl

hint: "next", "previous", "repeat", "list ingredients", "list containers", "go back", "restart"

Cook Said: "Alexa select one"
"Alexa go to thirty one"

The cook is unaware he is in a multithreaded recipe until he goes to the next instruction. Currently he has been preparing the rhubarb and has just transferred it to the oven and set a timer for 30 minutes.



Rice pudding with roasted rhubarb and tarragon - COMPLETE RECIPE
Cooking Instructions - 1 of 9 (Fixed Scale: 1)

Rhubarb

Check the rhubarb is cooked. If so remove from the oven and set aside to cool

Once it has cooled remove the tarragon sprigs and vanilla pod. As a suggestion, the vanilla pod can be reused to flavour caster sugar.

Main

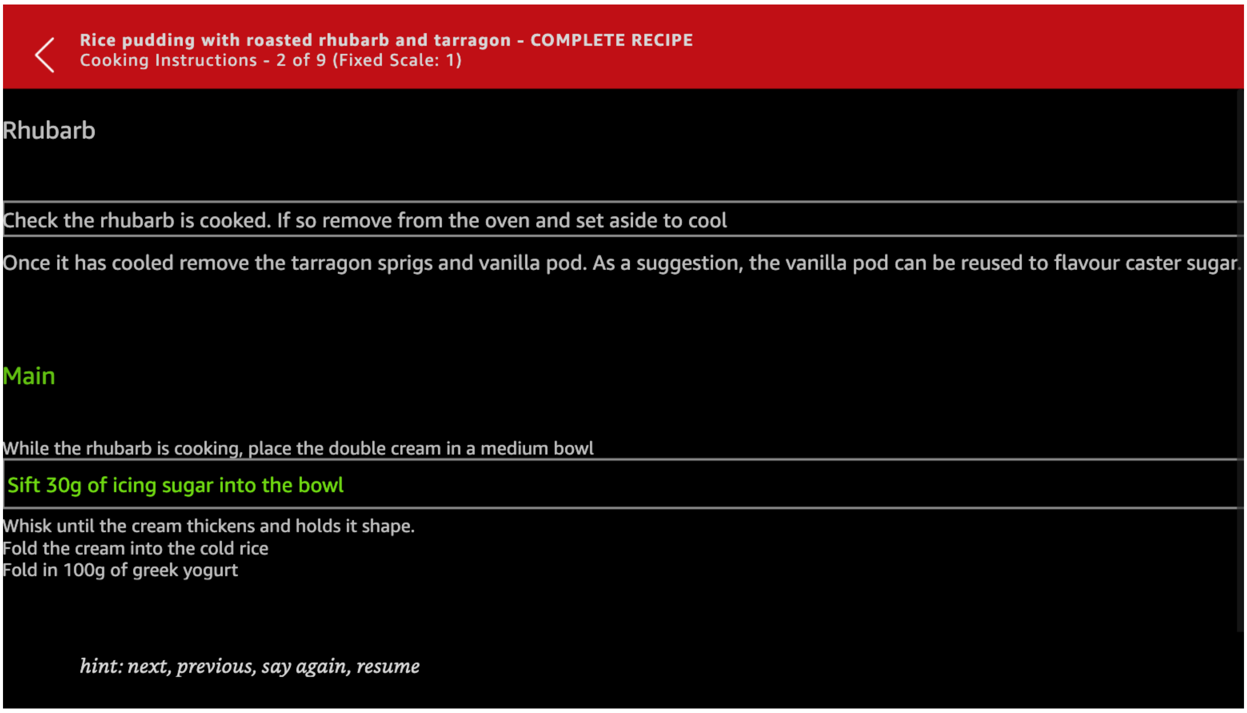
While the rhubarb is cooking, place the double cream in a medium bowl

Sift 30g of icing sugar into the bowl
Whisk until the cream thickens and holds it shape.
Fold the cream into the cold rice

hint: next, previous, say again, resume

Cook Said: "Alexa next"

Above is the multithreaded recipe screen. It is displayed in dark mode only because the colour green works best in this mode. The top half has a thread called Rhubarb (presently in oven) with the instructions required to compete that thread while the lower half shows the main thread containing the instructions to compete the remaining recipe. At this point the cook decides to run with it and continue with the recipe while the rhubarb cooks.



Rice pudding with roasted rhubarb and tarragon - COMPLETE RECIPE
Cooking Instructions - 2 of 9 (Fixed Scale: 1)

Rhubarb

Check the rhubarb is cooked. If so remove from the oven and set aside to cool

Once it has cooled remove the tarragon sprigs and vanilla pod. As a suggestion, the vanilla pod can be reused to flavour caster sugar.

Main

While the rhubarb is cooking, place the double cream in a medium bowl

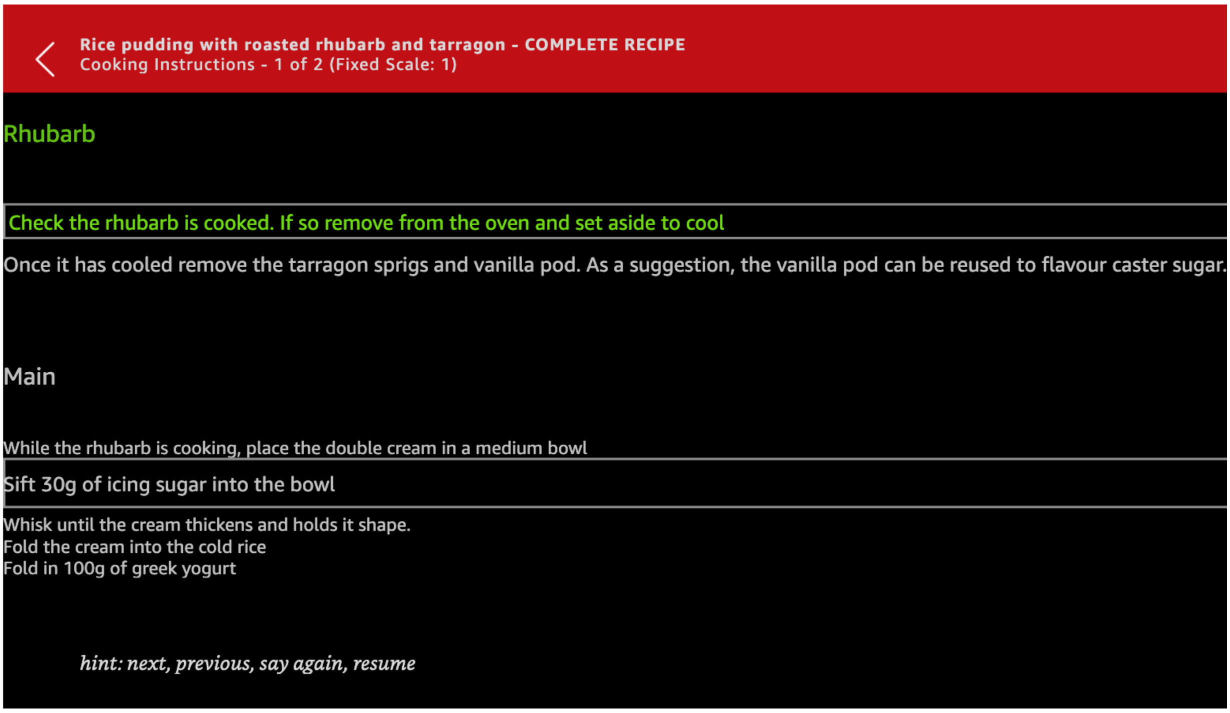
Sift 30g of icing sugar into the bowl

Whisk until the cream thickens and holds it shape.
Fold the cream into the cold rice
Fold in 100g of greek yogurt

hint: next, previous, say again, resume

Cook Said: "Alexa next"
"Alexa next"

Cook steps to the next instruction in the main thread. However immediately after competing this step he gets delayed by which time the rhubarb timer has gone. He then needs to complete the rhubarb portion of the recipe.



Rice pudding with roasted rhubarb and tarragon - COMPLETE RECIPE
Cooking Instructions - 1 of 2 (Fixed Scale: 1)

Rhubarb

Check the rhubarb is cooked. If so remove from the oven and set aside to cool

Once it has cooled remove the tarragon sprigs and vanilla pod. As a suggestion, the vanilla pod can be reused to flavour caster sugar.

Main

While the rhubarb is cooking, place the double cream in a medium bowl

Sift 30g of icing sugar into the bowl

Whisk until the cream thickens and holds it shape.
Fold the cream into the cold rice
Fold in 100g of greek yogurt

hint: next, previous, say again, resume

Cook Said: "Alexa resume"

Cook has resumed the rhubarb thread. At this point he can complete the entire recipe by saying "Alexa next" after completing each instruction. Alternatively, he can at any stage in the rhubarb thread say "resume" and Alexa will take him back to the main thread. He can then keep resuming between the two threads all he likes until the rhubarb thread is completed and then finally resumes to the main and completes it.