Both my father and mother are away from the castle, so I'm very busy!

I thought you'd be lonely. That's why I am here.

You can't mean your aide is more dependable than a prince!

Prince Raminess!

Eek!

Bang!!

HA HA HA!

Oh, beautiful Princess.

What's wrong?

No, Cain is here with me.

Awkward Fidget.

Hm? CAIN?

RAAAM!!!NEEEESSS!!

BANG!!

BANG!!

FLOOMP
You promised to go out on a date with me today.

No... uhm... well...

This is for you.

Gee, who is she?

What about my flowers?

Princess Ruruna, I'll be back...

Drag, drag...

Phew!

He is so annoying.

Good morning!

Gee, who is she?

Don't I know it...

But Raminess is the prince of the neighboring country.

You shouldn't treat him so lightly.

Good morning!
GEE! HOW LONG HAVE YOU BEEN THERE?

SINCE YOU OPENED THE BOOK A LITTLE WHILE AGO.

HELLO!

IT'S NOT A GOOD MORNING AT ALL. YOU ARE SO CAREFREE.

YOU SURPRISE ME EVERY TIME.

BUT THAT'S OKAY. ANYWAY, RAMINESS LEFT.

LET'S START TO DESIGN A DATABASE.

WAIT A MINUTE.

YOU CAN'T DESIGN A DATABASE WITHOUT ANY KNOWLEDGE.

A GOOD FOUNDATION IS IMPORTANT.

Yeah...

SQUEEZE SQUEEZE

STAY HERE.

HOP!

FIRST OF ALL, LET'S LEARN DATABASE TERMS.

AWWW...

LET'S DO OUR BEST.

DIVING INTO...
What Is a Relational Database?

What did you say?

N... No, nothing...

This is one of the files you are using.

Uh-huh.

A piece of data in the file is called a record.

And each item in the record is called a field.

One product is one record, right?

Uh-huh.

Then the product code, product name, unit price, and remarks are fields, correct?

<table>
<thead>
<tr>
<th>PRODUCT CODE</th>
<th>PRODUCT NAME</th>
<th>UNIT PRICE</th>
<th>REMARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>Melon</td>
<td>800G</td>
<td>With seeds</td>
</tr>
<tr>
<td>102</td>
<td>Strawberry</td>
<td>150G</td>
<td></td>
</tr>
<tr>
<td>103</td>
<td>Apple</td>
<td>120G</td>
<td></td>
</tr>
<tr>
<td>104</td>
<td>Lemon</td>
<td>200G</td>
<td></td>
</tr>
<tr>
<td>201</td>
<td>Chestnut</td>
<td>100G</td>
<td></td>
</tr>
<tr>
<td>202</td>
<td>Persimmon</td>
<td>160G</td>
<td></td>
</tr>
</tbody>
</table>
Each record contains fields of the same type.

<table>
<thead>
<tr>
<th>PRODUCT CODE</th>
<th>PRODUCT NAME</th>
<th>UNIT PRICE</th>
<th>REMARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 1</td>
<td>MELON</td>
<td>800G</td>
<td>WITH SEEDS</td>
</tr>
<tr>
<td>1 0 2</td>
<td>STRAWBERRY</td>
<td>1500G</td>
<td></td>
</tr>
<tr>
<td>1 0 3</td>
<td>APPLE</td>
<td>1200G</td>
<td></td>
</tr>
<tr>
<td>1 0 4</td>
<td>LEMON</td>
<td>200G</td>
<td>SOUR</td>
</tr>
<tr>
<td>2 0 1</td>
<td>CHESTNUT</td>
<td>100G</td>
<td>WITH BUR</td>
</tr>
<tr>
<td>2 0 2</td>
<td>PERSIMMON</td>
<td>1600G</td>
<td></td>
</tr>
<tr>
<td>3 0 1</td>
<td>PEACH</td>
<td>1300G</td>
<td></td>
</tr>
<tr>
<td>3 0 2</td>
<td>KIWI</td>
<td>200G</td>
<td>VALUABLE</td>
</tr>
</tbody>
</table>

For example, product code is a three-digit value...

AND product name is ten characters or less.

Then, next, let’s think about the product code in a little more detail.
What Is a Relational Database?

No two product codes are the same.

Yes. And there are no duplicated records. So, if you know the product code is 101, you can identify it as melon.

That makes sense.

I understand.

But in the case of unit price...

What will happen?

Oh, I found the same number.

Even if you know a fruit's price is 200G...

You will not know whether it indicates lemon or kiwi.