

RE: How do I handle this??

Source:

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.tablesdbdesign/2008-05/msg00100.html>

- *From:* Duane Hookom <duanehookom@xxxxxxxxxxxxxxxxxxxx>
 - *Date:* Mon, 5 May 2008 09:02:01 -0700
-

A primary key field must contain unique values (no duplicates). My sentence following the ProjectIssues table structure states why this would be an issue.

Duane Hookom
Microsoft Access MVP

"mauricio" wrote:

Ok, that helps!

Can you explain why it is better to have ProjIssID (autonumber) as the primary key rather than using the ProjectID again (as in the Projects Table)

Thanks

"Duane Hookom" wrote:

I am fairly certain I suggested how to structure your project updates table. I provided at least three fields that I would place in the table. I also made a comment "don't know what you want to store..." but you only stated "store weekly updates" which has no specific meaning. Do you want to store the date of the update? How about who is entering the record? Do you want comments regarding the project?

Assuming you have a Project table with one record per project and a primary key of ProjectID. Your project issues table might look something like:

ProjectIssues

=====

ProjIssID autonumber primary key
ProjectID numeric long links to Projects.ProjectID
IssueTitle shorter text highlighting the main point of the issue
IssueDescription memo field with longer description of issue
IssueStatus has this been resolved or terminated or what

RE: How do I handle this??

A project could have zero to dozens of records in the ProjectIssues table. This would be similar to your other tables such as the ProjectResources table.

--

Duane Hookom
Microsoft Access MVP

"mauricio" wrote:

The ProjectUpdates table is supposed to store weekly updates for each project. There might not be update on certain weeks. What's another (better) way to create this table??

As far as the other tables having the same primary key... This is where I need help... Would it be better to create 1 big table that has project details, project resources, project issues, project updates??

"Duane Hookom" wrote:

It looks like 5 of the 6 tables you mention have the same primary key. I don't recall ever creating an application where the same field is used as the primary key of more than a single table.

I don't know what you want to store in project updates. I would expect a ProjectID, Update, UpdateComments,... Each of these records would store only one date. There would potentially be 52 (more or less) records with the same ProjectID.

You should check out <http://www.simple-talk.com/content/article.aspx?article=354> as well as other resources regarding normalization and database design.

RE: How do I handle this??

RE: How do I handle this??

--

Duane Hookom
Microsoft Access MVP

"mauricio" wrote:

Thanks Duane,

Did you see the 2nd part of my post where I gave the details of each table.

I am not sure how to allow the application to create 52 records for the updates, can you please expand a little so I can understand.

Thanks,
Mauricio

"Duane Hookom" wrote:

You haven't provided much detail on fields, primary and foreign keys, and relationships. What you did provide ProjectID update1 update2.....update52 seems very wrong IMHO. I would allow the application to create 52 records if needed rather than 52 update fields. You

RE: How do I handle this??

are creating
a
spreadsheet,
not a
relational
database.

--
Duane
Hookom
Microsoft
Access
MVP

"mauricio"
wrote:

I
also
have
another
question.
This
is
my
basic
database
design
and
I
want
to
see
if
I
am
taking
the
right
approach.

1.
Projects
Table:
Main
table,
contains
start
date,

RE: How do I handle this??

RE: How do I handle this??

end
date,
projmgr,
etc.

–
details
about
project
(key:
projectid)

2.
Project
Ranking:
Project
Ranking
in
6
categories
and
total
rank
score
(key:
projectid)

3.
Project
Resources:
Which
resources
are
assigned
to
this
project
(key:
projectid)

4.
Project
Updates:
Weekly
update
for
each
project
(key:
projectid)

5.

RE: How do I handle this??

Project
Issues:
Issues
for
each
project
and
details
like
date,
owner,
etc
(key:
projectid)

6.
Employees:
A
list
of
all
employees
who
work
on
the
company
and
could
be
the
resources
assigned
to
a
project.
(key:
employee
id)

Let
me
have
it!!!
How
badly
designed
is
this
database???
I

RE: How do I handle this??

RE: How do I handle this??

also
don't
know
how
to
relate
the
tables....
Please
help!

"mauricio"
wrote:

I
have
a
"projects"
database.
I
need
a
UpdateStatus
field
on
my
table
so
the
users
can
enter
weekly
updates
on
their
projects.
I
need
to
be
able
to
keep
these
updates
for
52
weeks.

RE: How do I handle this??

RE: How do I handle this??

I
also
need
to
timestamp
their
entries.
Not
every
project
will
have
an
update
everyweek.

I
created
a
table
like
this.
(I
already
have
a
Projects
Table,
with
all
other
details
about
the
project,
ProjectID
is
the
key
for
that
table)

ProjectID
update1
update2.....update52

Is
this
the
best

RE: How do I handle this??

RE: How do I handle this??

design
for
this??
If
it
is,
how
do
I
get
the
timestamp
for
each
entry?