Assignment Number:	INN210.4
Assignment Name:	Access Assignment – Part A
Weighting:	7%.
Due Date:	Friday, 26th September 2014 Rather than leaving it to the last minute, you are strongly encouraged to start working on this assignment at a much earlier stage. The relevant material on Microsoft Access is covered in Lectures 7 and 8, and in the related Practicals.
Items to be Submitted:	**AccessAssign142_Oly2012.accdb** – *You must upload this specific file.* DO NOT put your database inside a zip file, or inside any other kind of file. If you ignore this requirement, you will receive zero. Make sure that Microsoft Access is CLOSED when you upload your database to Blackboard. Otherwise, your database will very likely be uploaded in a corrupted state. If your uploaded database is corrupted, then you will very likely receive a lower mark.
How to be Submitted:	The assignment must be submitted using Blackboard.

Introduction to the Access Assignment (Parts A, B and C)

Overview

The Access Assignment is designed to build on the database concepts that you have already covered in INN210, and give you experience in using the **Rapid Application Development** features of the widely-used database software, Microsoft Access 2010.

Together, the three assignment parts (A, B and C) will help you to learn how to develop forms and reports: key ingredients in any real-world application.

Part A is the foundation for Parts B and C. That is, Part A must be done *before* attempting either of the other parts. Because of this foundational role, Part A is meant to be relatively easy. Parts B and C will be a bit more challenging.

Key Approach

When implementing a form or report, it is important to *get it working properly first*, and *fix up the appearance later*. There is a major reason for this. When working on the layout of a form/report, *it can take quite a while to get all the controls and their labels in the right place*, with the right size, fonts, etc. But there can be times when, having made a first attempt at creating a form/report, you decide that it's best to throw it away and start again, rather than trying to patch up the mess that you've already got.

In other words, often when we try to do something, we don't always succeed first time. But, by trying, we learn more about what we're supposed to be doing, and are more successful at a second, or third, attempt. But if you start again from scratch, you then have to redo all the same work getting all the controls in the right place, etc.

In this regard, the Adamski & Finnegan textbook sometimes sets a bad example. In some of their exercises, they get you to start with a form/report created by a Wizard, then fixup the layout straight-away, before adding extra features. This works in the textbook, because the authors know what to, and never have to throw anything away. You're not quite in the same situation.

So, in this assignment (Parts A through C), you are asked to *get it working first*, and then to *fix up the appearance later*.

NB: Compact and Repair

1. You are advised to use **Compact and Repair** on a regular basis – see textbook pages AC 38-40. (Click the **File** tab in the Ribbon, make sure that the **Info** tab is selected and then click **Compact & Repair Database**.) When developing your application, the database file can become disorganised and this can cause Access to be unable to work with your database. Using **Compact and Repair** will help ensure you don't lose any work because of this.

   Also, you would be wise to take a backup copy of your database before you make any major changes.

2. When you've completed your assignment, you may wish to *improve your upload speed* to Blackboard, especially if you are off-campus. Using **Compact and Repair** will make your database smaller and faster to upload.

   Beware: if you add background pictures for your report/form/subform or other stylised techniques such as colouring or shadow effects these tend to make the size of the database bigger. It is recommended *not* to use such effects in this assignment, and there are no additional marks for doing so. If you experiment with such features, it is recommended that you remove them before doing a final **Compact and Repair** and then submitting your assignment.

Do's and Don'ts

(a) When you are working on your assignment, *all of your work in stored inside your database file*. Think about what would happen if you had only one copy of that file and you lost it before submitting, e.g. by accidentally losing your USB-stick.

Take regular backups, but make sure that Microsoft Access is *closed* when you do this. Otherwise, your backup database(s) will very likely be in a corrupted state. I.e. no good as backups.

(b) You are reminded that assignments in INN210 are **not group work**. See note (a) for SQL Assignment.

(c) **You must use Access 2010 to do all the parts of this assignment**. Don't try to use an older version of Access, such as Access 2007 or 2013, to do this assignment. There are two reasons for this. Firstly, some parts of this assignment specification might not make sense with Access 2007 or 2013.

Secondly, an Access 2007 or 2013 database might not be entirely compatible with Access 2010. While it is sometimes possible to convert a database between Access 2010 and Access 2007 or 2013, experience shows that some things often *fail to work properly*, after such conversions.

The bottom-line is this. If you convert your assignment from one version of Access to another, and it then doesn't work properly, you will lose marks.

(d) Don't delete the **CopyFormAndSubform** form or the **CopyRelationships** form, which is part of the database that you download from Blackboard. Although you won't use the first of these forms, **CopyFormAndSubform,** in this Part of the assignment, you'll need it when you start the next part, **Part B**. If you need to recreate your database, as described in the Appendix, then you may need the second form, **CopyRelationships**.

(e) The supplied database uses **overlapping windows**, rather than **tabbed documents**. Do ***not*** change this setting.

(f) In *some* parts of the following, references to relevant pages in the **textbook**, Adamski & Finnegan's Access 2010, are provided. That material is also covered in Practicals, so these references are just meant to make it easier for you to find this material again. While the textbook material is a help, to get a good mark *you still have to think about what you're doing*, and take the appropriate actions, rather than following the textbook *blindly*, at each step. For example, given the requirements of this assignment below, doing *part* of step 8 on page 95 is appropriate and *part is not*.

(g) **Do *not* use nested queries** in *any part* of the assignment. In other words, do ***not*** use a query in the FROM clause of another query.

(h) **You must use the names given** in the assignment specifications or in the supplied database. That is, you *must* name your queries, reports, forms, or subforms as specified below, and you *must not* change the names of any tables, or fields within tables, in the supplied database. Failure to use these names will result in loss of marks, i.e. using different names for a query, report, form, or subform, or changing the name of any of the tables or fields.

(i) You may add, change, or delete rows in the supplied database, **but do *not* change its structure**. I.e. don't add or delete columns, and don't change the datatypes of existing columns. Likewise, do not add new tables.

(j) If you make some changes to the rows in the supplied tables and then later want to return to the original rows, without having to redo the whole assignment from scratch, see the Appendix, **Moving Your Work to Another Copy of the Database** (at the end of this document).

As with the SQL assignments, be careful about the following.

(k) *When writing your queries you must not assume that particular rows are present (or absent)* in the database. That is, your queries *must still give appropriate results, even if all the current rows were deleted and a completely different set of rows inserted*, e.g. if the data were replaced with data concerning different competitors, events, etc.

(l) The data in the database can be used to help you develop your answers, *but there is no requirement to use the given rows in that database, as it is given to you*. In particular, there is no guarantee that the data in that database will be sufficient to test whether your answers always work correctly or not.

The Database

You are to use the **AccessAssign142_Oly2012.accdb** database, for all three parts of the Access Assignment. An initial version of this database is on Blackboard.

This database provides tables that are similar to those used in the SQL Assignment, except that the MedalMatrix table has been added. See page 115 of the **Lecture Notes** PDF for a description of this table. Although the table supplied in this Access database has 8 rows, rather than the 3 rows in the **Lecture Notes**, it still works the same.

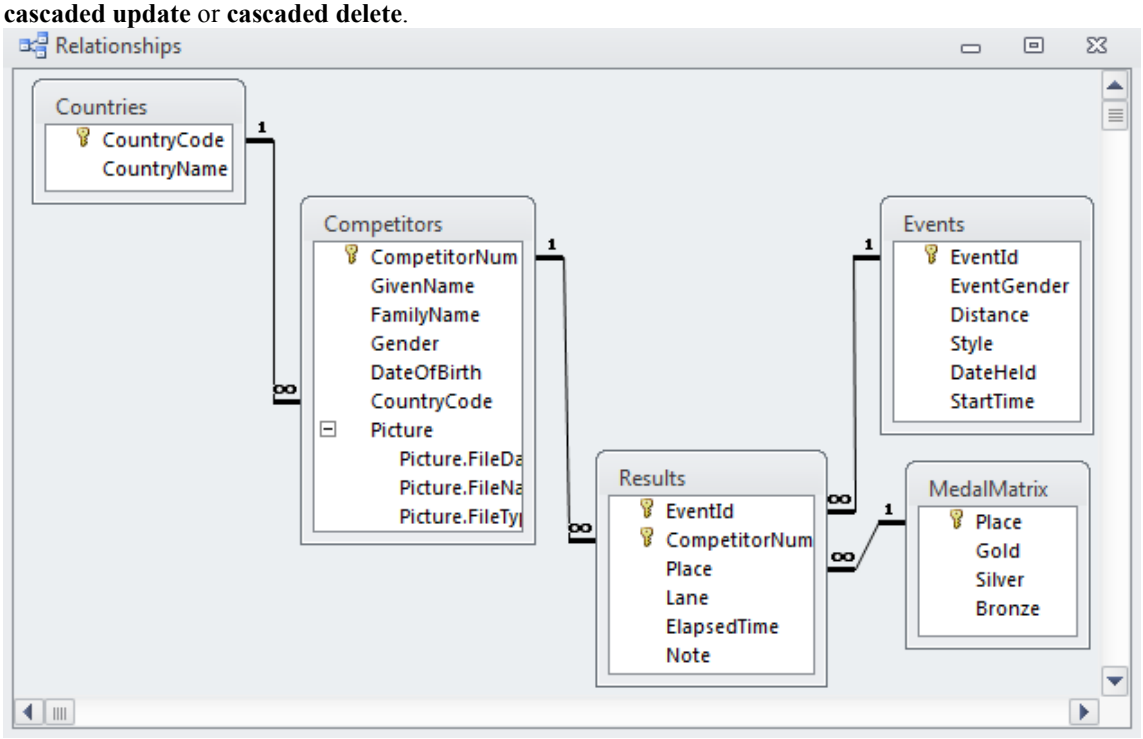
Task 1

In Access, use the **Relationships Window** to create the database's foreign-key relationships as shown in the screen-snapshot below. (Reminder: textbook pages AC 92-96.) Note that:

(a) This is essentially a Relational Map, as you've seen when studying SQL. In Access, each **primary-key** to **foreign-key** relationship – i.e. each **one-to-many** relationship – is shown as a line with **1** and ∞ symbols at each end, rather than a single line at one end and a crows-foot at the other.

(b) As far as this assignment is concerned, *the important thing is* that you create *each of the four relationships* shown below. The *positions* of the boxes for each table, and their *sizes*, *will not affect your marks*. Also, if some table boxes are displayed with scroll-bars, you may like to remove these by dragging the border of each such box to enlarge it, but you don't have to.

(c) When you create each relationship, ensure that **Referential Integrity** is enforced, but *don't enable* **cascaded update** or **cascaded delete**.



Important: do not attempt the following tasks until you have completed Task 1. Doing that task makes it a lot easier to do each of the remaining tasks.

Task 2

In tasks 2 and 3, you are to start developing a form that shows the results in each swimming event, by first creating a query, and then creating a form that gets some of its data from that query.

For this task, create a query – named **ResultsWithNames** – that produces the kind of output shown in the screen-snapshot below, and fulfils the requirements stated below. (Reminder: textbook pages AC 118-119.)

ResultsWithNames

EventId	Place	GivenName	FamilyName	CountryCode	Lane	ElapsedTime
SWM010	1	Florent	Manaudou	FRA	7	21.34
SWM010	2	Cullen	Jones	USA	5	21.54
SWM010	3	Cesar	Cielo	BRA	4	21.59
SWM010	4	Bruno	Fratus	BRA	6	21.61
SWM010	5	Anthony	Ervin	USA	3	21.78
SWM010	6	Roland	Schoeman	RSA	8	21.80
SWM010	7	George Richar	Bovell	TRI	2	21.82
SWM010	8	Eamon	Sullivan	AUS	1	21.98
SWM011	1	Nathan	Adrian	USA	5	47.52
SWM011	2	James	Magnussen	AUS	4	47.53
SWM011	3	Brent	Hayden	CAN	7	47.80
SWM011	4	Yannick	Agnel	FRA	1	47.84
SWM011	5	Sebastiaan	Verschuren	NED	6	47.88
SWM011	6	Cesar	Cielo	BRA	2	47.92
SWM011	7	Hanser	Garcia	CUB	3	48.04
SWM011	8	Nikita	Lobintsev	RUS	8	48.44
SWM012	1	Yannick	Agnel	FRA	5	103.14
SWM012	2	Yang	Sun	CHN	4	104.93
SWM012	2	Taehwan	Park	KOR	3	104.93
SWM012	4	Ryan	Lochte	USA	2	105.04
SWM012	5	Paul	Biedermann	GER	6	105.53
SWM012	6	Robbie	Renwick	GBR	1	106.53

Record: 1 of 208 No Filter Search

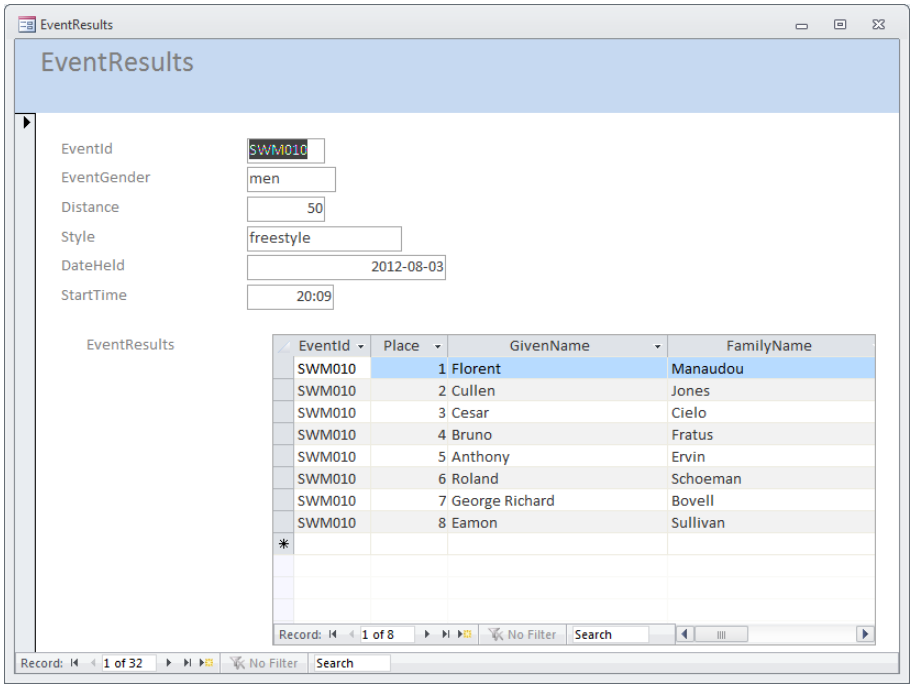
The purpose of the query is to display data from the **Results** table, but with the columns in a different left-to-right sequence, and with each competitor's name and country-code also displayed. Your query must sort the the displayed rows by **EventId** overall, and then by **Place** within each event.

Specific Hints and Warnings for this Task:

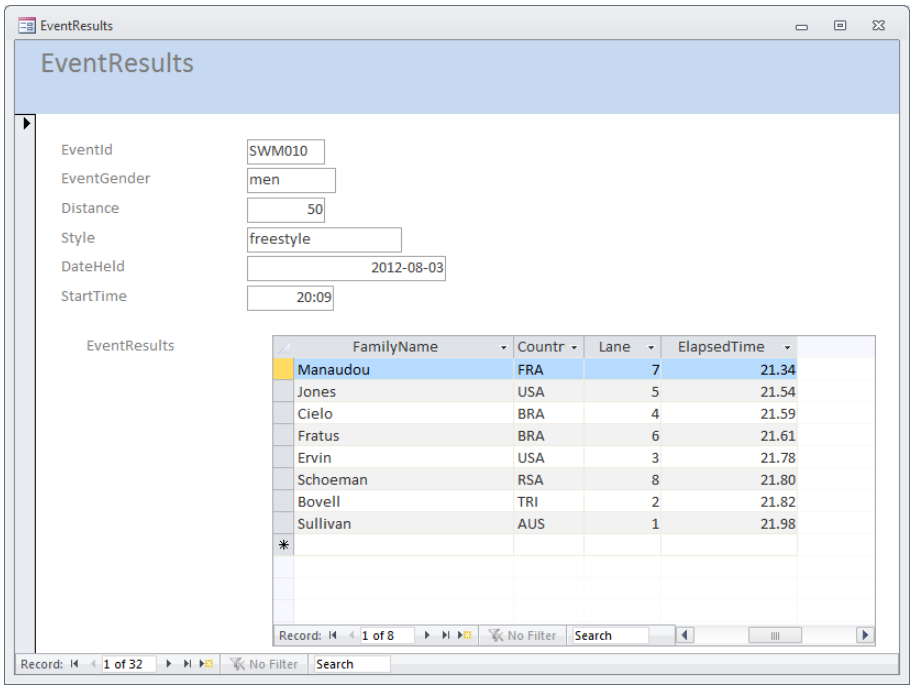
(a) Use the *correct name* for your query.
(b) While you are permitted to use **SQL View**, you are encouraged to use **Design View** to create your query, because it makes life easier by automatically creating the joins for you (assuming that you have done Task 1 above). **Design View** is also known as **QBE** (Query By Example) – see Lecture 8.
(c) Before creating your query, think about *which tables are required* – i.e. where is the data coming from?
(d) Be careful *not to include any tables that your query doesn't really need*. For example, you might think that you need the **Events** table, because you need the **EventId** column. But that would only be true *if the **Events** table was the only one* that included the **EventId** column.
(e) The column-widths shown above were adjusted so that you can clearly see the required name of each displayed column, but the exact widths are unimportant in your assignment. I.e. won't affect your marks.
(f) The displayed columns *must have the same names as shown above*, e.g. **EventId** and **Place**.
(g) Don't use:
- subqueries;
- group by;
- a filter (ref: textbook page AC 127);
- nested queries, within your query. In other words, you ***must not*** use a query in the FROM clause of another query.

Task 3

While taking note of the **Specific Hints and Warnings** below, **c**reate the following form, by using the **Form Wizard**. (Reminder: textbook pages AC 188-191.) The purpose is to show information about each event (one-at-a-time) on the main form, and details of the related results on the subform. In this task, the *visual appearance of this form is deliberately left in a rough state*. You will be asked to enhance it substantially in **Part B**, the second part of this Access Assignment.



The following figure shows the same **EventResults** form, but with the bottom scroll-bar moved more to the right, so that *you can see all the columns in the subform*.



Specific Hints and Warnings for this Task:

(a) Use the *correct* names for your form and subform. The main form must be named **EventResults** and the subform must be named **EventResults Subform.**

cont...

(b) When using the **Form Wizard**, use the **Events** table as the source of data for your main form, and the above query, **ResultsWithNames,** as the source of data for your subform.
(c) When the Wizard asks you which columns/fields do you want from the above two sources, *include them all*.
(d) Don't try to use linked forms.
(e) When the Wizard asks you what layout you would like for your subform, choose **Datasheet**, not **Tabular**.
(f) When Access opens a form, it takes the size of your computer's screen into account, so *do not be concerned if your form is a bit bigger or smaller than the snapshots shown above*.
(g) If you have studied Access before doing INN210, and regard yourself as either "an expert who doesn't want to use the Form Wizard", or "an expert who knows how to fix things after using the Form Wizard", please note the following additional warnings. (If you use the Form Wizard, you won't even have to think about these things.)
   i. You *must* use a subform on this form.
   ii. Don't try to use anything else, such as an embedded query.

Task 4

In tasks 4 and 5, you are to start developing such a report about the medal winners, by first creating a query, and then creating a report that gets its data from that query.

Create a query – named **MedalWinners** – that produces the kind of output shown in the screen-snapshot below, and fulfils the requirements stated below. (Reminder: textbook pages AC 118-119, 130-145, 152-154.)

MedalWinners

CompetitorNum	CountryName	FamilyName	GivenName	SumOfGold	SumOfSilver	SumOfBronze
1016284	Russian Federatio	Efimova	Iuliia	0	0	1
1016285	Russian Federatio	Zueva	Anastasia	0	1	0
1018529	Russian Federatio	Korotyshkin	Evgeny	0	1	0
1026870	Japan	Irie	Ryosuke	0	1	1
1026880	Japan	Tateishi	Ryo	0	0	1
1026915	Japan	Matsuda	Takeshi	0	0	1
1026921	Japan	Hagino	Kosuke	0	0	1
1029075	Japan	Terakawa	Aya	0	0	1
1029083	Japan	Suzuki	Satomi	0	1	1
1029113	Japan	Hoshi	Natsumi	0	0	1
1043098	Belarus	Herasimenia	Aliaksandra	0	2	0
1044790	Tunisia	Mellouli	Oussama	0	0	1
1056740	South Africa	Van Der Burgh	Cameron	1	0	0
1061698	South Africa	Le Clos	Chad	1	1	0
1062467	Great Britain	Adlington	Rebecca	0	0	2
1065297	Lithuania	Meilutyte	Ruta	1	0	0
1072444	People's Republic	Ye	Shiwen	2	0	0
1072467	People's Republic	Jiao	Liuyang	1	0	0
1072471	People's Republic	Li	Xuanxu	0	0	1
1072481	People's Republic	Lu	Ying	0	1	0
1072491	People's Republic	Tang	Yi	0	0	1
1072528	People's Republic	Sun	Yang	2	1	0

Record: 1 of 57 No Filter Search

The purpose of the query is to display information about people who have won at least one gold, silver, or bronze medal.

More specifically, there is one row displayed for each such person. The three right-hand columns show the total number of gold, silver and bronze medals won by the person. E.g. Sun Yang (last row above) won 2 gold, 1 silver and no bronze medals (as far as this database is concerned). People who didn't win any medals do not appear.

Specific Hints and Warnings for this Task:

(a) Use the *correct* name for your query.
(b) As before, you are encouraged to use **Design View** to create your query.
(c) Before creating your query, think carefully about *which tables are required* – i.e. where is the data coming from – and any conditions/criteria in your query.

(d) Remember point (g) on page 2, and *do not use your earlier query*, **ResultsWithNames**, in this query.
(e) The comments in Task 2 about column-widths, and column names, also apply here, e.g. **SumOfGold**.
(f) If you use SQL to write this query, you'll need a "group by". If you use Query Design View, you'll need the **Total** row (ref: textbook page AC 151).
(g) Don't sort the rows. If your query produces the same rows as above, but in a different order, that's fine.
(h) Don't use:
- Subqueries;
- a filter (ref: textbook page AC 127);
- nested queries, within your query. In other words, you ***must* not** use a query in the FROM clause of another query.

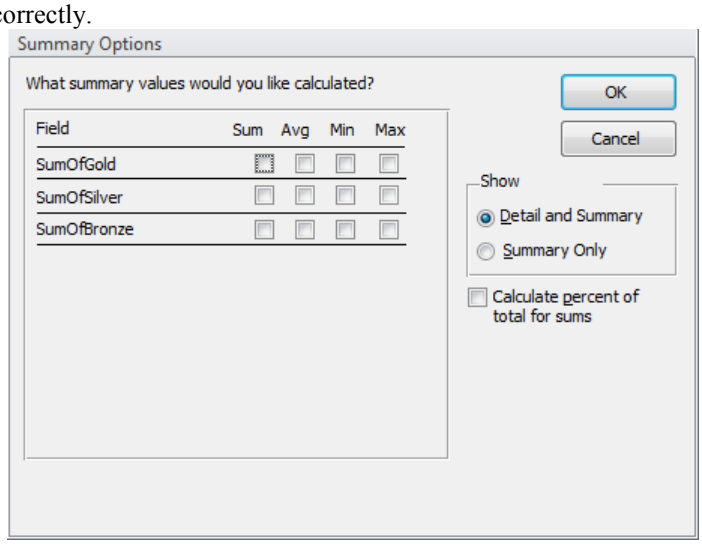
Task 5

Based on the above query, use the **Report Wizard** to create the report, also named **MedalWinners**, shown on the following page. (Reminder: textbook pages AC 186-187, 192-196.)

To improve the *user-friendliness*, this report presents the results of your **MedalWinners** query in a better format, by grouping the competitors together by country. As part of this, *the report sorts the data* in ascending order of **CountryName**, **FamilyName** and **GivenName**, even though *your underlying query does not sort the data*.

Specific Hints and Warnings for this Task:

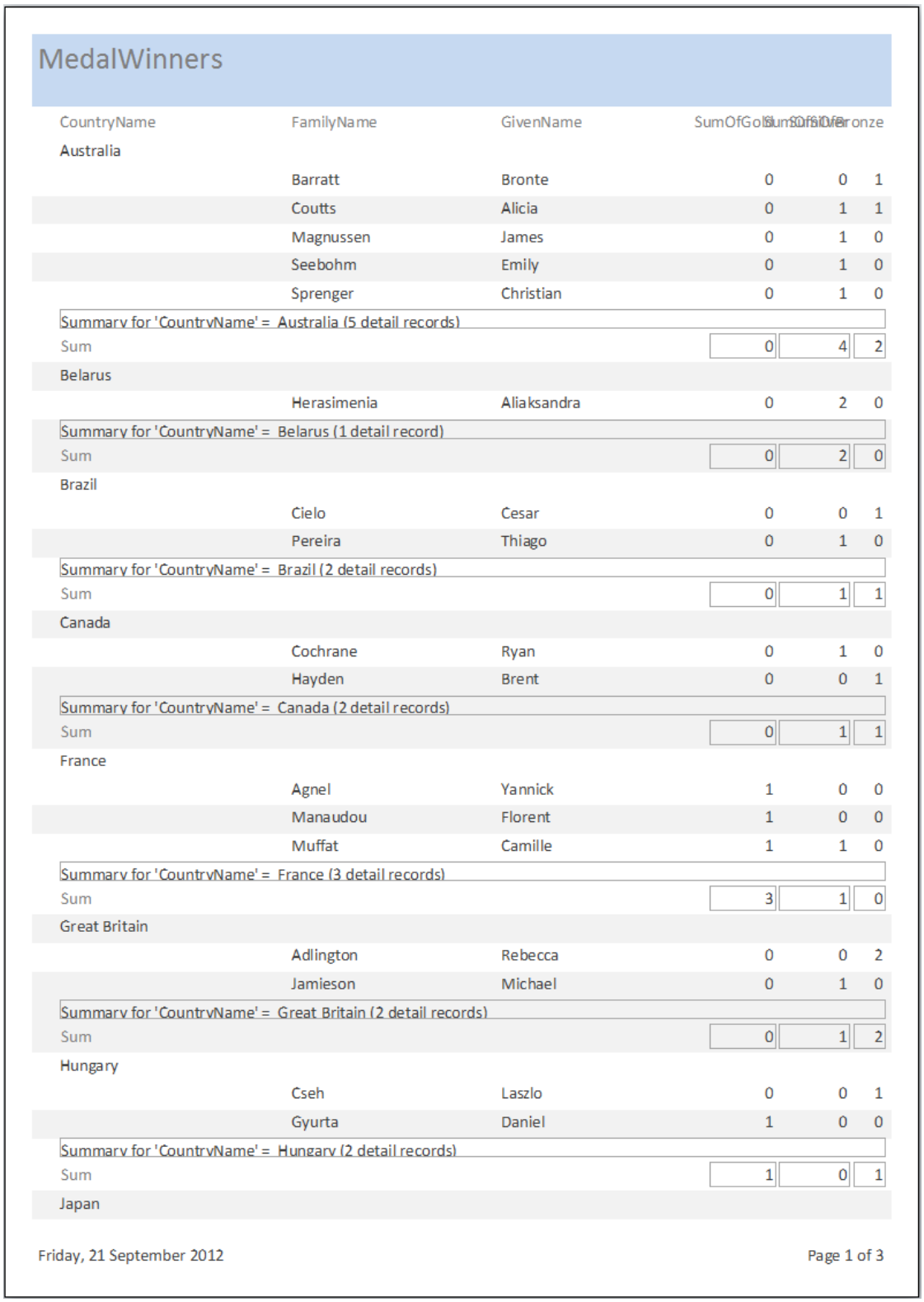
(a) Use the *correct* name for your report.
(b) Don't change your query from task 4, when doing task 5. In particular, *do not try to sort the rows in your query*. The report will do that for you.
(c) Use your query as the report's source of data (**Tables/Queries** in the **Wizard**), rather than any of the tables in the database.
(d) When the Wizard asks you whether you want to add any grouping levels, choose **CountryName**, so that the competitors are grouped together (and sorted) by country.
(e) When you get to the step where the Wizard asks you **What sort order and summary information do you want for detail records?**,
   - tell it that you want the report to be sorted by **FamilyName** and then **GivenName**; and
   - investigate the use of the **Summary Options...** button, which leads to the dialog box shown below. You will need to complete this dialog box, before proceeding. While this is a useful feature, be warned that it isn't well documented. If you can't figure out which options you should select to generate the report shown on the next page, then just try one or two, and be prepared to re-run the **Report Wizard** from the start, if you guess incorrectly.



(f) To make your report have the same visual appearance as the one on the next page, use **Stepped** as the layout, and **Portrait** as the Orientation.
(g) Do *not* try to use the Hide Duplicates property (ref: textbook AC 392).
(h) Once again, if you have studied Access before doing INN210, and regard yourself as either "an expert who doesn't want to use the **Report Wizard**", or "an expert who knows how to fix things after using the **Report Wizard**", please note the following additional warnings. (If you use the **Report Wizard**, you won't even have to think about these things.)
   i. Do *not* create an SQL statement for the RecordSource that uses your **MedalWinners** query in the From clause.
   ii. You must use Access's **Grouping and Sorting** feature for this report.
   iii. You must *not* use a subreport in this report.

The following snapshot shows the first page of the report, in **Print Preview** mode – the mode you normally see when the **Report Wizard** finishes. The date shown at the bottom of the page will change, according to when you run the report. *Using different computers can influence the precise layout and number of pages in the report, so do not worry about that (in this part of the assignment).* Also, changing the rows in your copy of the database may make the data on your report differ from that shown here.

The **Report Wizard** doesn't do a perfect job, with some overlapping of the column headings shown below. *For now, the layout of this report is deliberately left in a rough state.* You will have an opportunity to fix these problems and make other enhancements to this report, in **Part C** of this assignment.

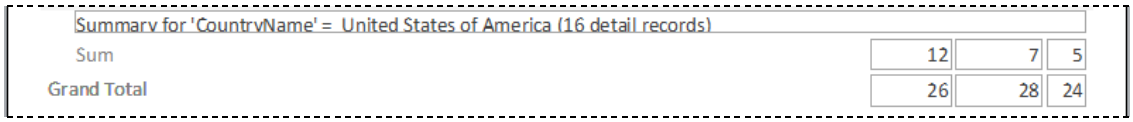


MedalWinners

CountryName	FamilyName	GivenName	SumOfGold	SumOfSilver	SumOfBronze
Australia					
	Barratt	Bronte	0	0	1
	Coutts	Alicia	0	1	1
	Magnussen	James	0	1	0
	Seebohm	Emily	0	1	0
	Sprenger	Christian	0	1	0
Summary for 'CountryName' = Australia (5 detail records)					
Sum			0	4	2
Belarus					
	Herasimenia	Aliaksandra	0	2	0
Summary for 'CountryName' = Belarus (1 detail record)					
Sum			0	2	0
Brazil					
	Cielo	Cesar	0	0	1
	Pereira	Thiago	0	1	0
Summary for 'CountryName' = Brazil (2 detail records)					
Sum			0	1	1
Canada					
	Cochrane	Ryan	0	1	0
	Hayden	Brent	0	0	1
Summary for 'CountryName' = Canada (2 detail records)					
Sum			0	1	1
France					
	Agnel	Yannick	1	0	0
	Manaudou	Florent	1	0	0
	Muffat	Camille	1	1	0
Summary for 'CountryName' = France (3 detail records)					
Sum			3	1	0
Great Britain					
	Adlington	Rebecca	0	0	2
	Jamieson	Michael	0	1	0
Summary for 'CountryName' = Great Britain (2 detail records)					
Sum			0	1	2
Hungary					
	Cseh	Laszlo	0	0	1
	Gyurta	Daniel	1	0	0
Summary for 'CountryName' = Hungary (2 detail records)					
Sum			1	0	1
Japan					

Friday, 21 September 2012 Page 1 of 3

On the last page of this report, a **Grand Total** will also be shown. This is a total of each type of medal, across all the countries. For the supplied database, the last part of this report will appear something like the following screenshot.



Summary for 'CountryName' = United States of America (16 detail records)			
Sum	12	7	5
Grand Total	26	28	24

Reminders: *before submitting your assignment*:

(a) Check that you have met all the requirements under the **Do's and Don'ts** listed in **Part A**, and under *each* **Specific Hints and Warnings for this Task** section above. You don't want to lose marks for things that you think are unimportant, but which the marker views differently.

In particular, just because your answer "looks right" on the screen does ***not*** mean you will get full marks. As a simple example, if you use the wrong name(s) for your queries, form, or report, then you will lose marks.

(b) Remember to use the **PAT Add-in** to check whether your assignment is correct or not.

(c) Remember to use **Compact and Repair** just before you submit your database to Blackboard, so that your database is smaller and uploads more quickly. (Click the **File** tab in the Ribbon, make sure that the **Info** tab is selected and then click **Compact & Repair Database**.**)**

Distribution of Marks

Task 1: Foreign-key Relationships	1.00%
Task 2: Query	1.50%
Task 3: Form with subform	1.00%
Task 4: Query	2.00%
Task 5: Report	1.50%
Total Marks	**7.00%**

Part marks will be given for a partly correct solution. A late penalty will be deducted for late assignments, in accord with Faculty policy.

Appendix **Moving Your Work to Another Copy of the Database**

Having everything in a single (.accdb) file makes some things easier and some things harder. If you change or delete some of the rows in one or more of the tables, and want to restart with a fresh copy from the original database, this can be done if you're careful.

If you destroy some rows and haven't got very far in doing this assignment, then the easiest way is to start again with an entirely fresh copy of the supplied database. But, if you want to keep your queries/reports/etc, then here's a fairly easy way to do it.

Getting Started

1. If you're already in Access, exit from it.

2. Rename your existing database to the name **Old.accdb**. **NB: you must use this name.** Otherwise the following procedure won't work.

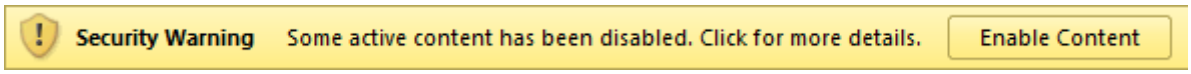
3. Get a new copy of the original database (from Blackboard, for example). **NB: put it in the same folder as your Old.accdb database.** Otherwise the following procedure won't work.

4. Open your new copy of the database.

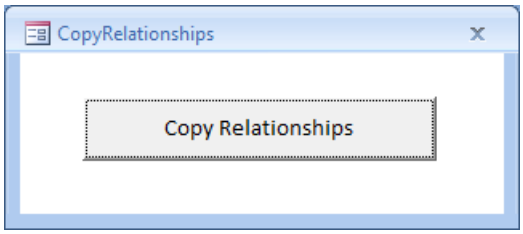
Copying Your Relationships

Copying relationships is tricky. While Access's Import facility allows you to import relationships when you import *all* the tables from your old database, that doesn't help when your old tables are damaged or some rows have been deleted. Because Access doesn't provide a simple way of *copying only your table relationships*, we give you an automated way of doing this.

5. To guard against viruses, Access 2010 disables executable code by default. Here, we need to enable that code. To do this, click **Enable Content** in the Message Bar, as shown below. All executable code is now allowed to run, in this specific database.



6. Open the **CopyRelationships** form (in **Form View**).



7. Click the **Copy Relationships** button on this form.
   If everything is ok, a **Done** message box will appear. Close that message box, and then close this form.
   If that **Done** message box *does not appear*, check that you have done steps 2, 3 and 5 correctly.

8. In the **Database Tools** tab, click on the Relationships button to open the **Relationships Window**. If that window is blank, right-click on it and select **Show All**.

9. All the tables should now be shown, together with their relationships. They might look a bit of a mess, because their original positions have been lost. If you like, you can drag the tables around to make their positions more like the ones given in **Task 1**. But you don't have to. The important thing is that the relationships exist. The positions of the tables is unimportant.

10. Close the **Relationships Window**. If you get prompted to **save changes**, click **Yes**.

Copying Your Queries/Forms/Reports

11. On the Ribbon, select the tab **External Data**, and then click the **Access** button in the **Import & Link** group. Browse to where your old database is located. With the **Import ...** radio button selected, click **OK**.

    The **Import Objects** dialog box appears.

12. Select the tab for the type of objects that you wish to copy from your old database (e.g. Queries). Then select the item(s) that you want to copy. *Don't click **Ok** just yet*.

13. Repeat step 12 for any other types of objects that you want, i.e. forms and/or reports.

14. Click on **OK**, to import all the objects you've selected.

15. When the **Save Import Steps** dialog box appears, just click **Close** (without saving the steps).

    The imported objects will now be accessible via the **Navigation Pane**.

16. Keep your old database in a safe place. You never know – you might want to use it again, if things go really badly.