Base de Datos

Definición:

Conjunto de datos almacenados estructuralmente mediante una computadora, de tal modo que pueden ser recuperados, interrelacionados y manipulados con facilidad.

Las bases de datos permiten el acceso de uno o varios usuarios a grandes cantidades de información de una forma rápida y racional. La posibilidad actual que tienen las grandes computadoras de acceder a grandes masas de memoria, la capacidad de diálogo interactivo, local o remoto, mediante terminales y la rapidez y facilidad de gestión de los sistemas operativos que permiten relacionar distintas partes de las bases hacen de las tecnologías de las bases de datos uno de los campos de la informática que tienen una mayor expansión. Toda base de datos implica la existencia de un sistema gestor que permite introducir estructuradamente la información y recuperarla y examinarla según diferentes criterios. Las aplicaciones más usuales de bases de datos están relacionadas con la gestión de grandes empresas e instituciones públicas. Otra aplicación importante es la informatización de bibliotecas y la creación de bancos especializados de datos.

Introducción:

El correcto planteamiento de una base de datos en sus fases iníciales, determinará en gran medida su éxito. Un análisis o planteamiento incorrecto en la fase inicial repercutirá negativamente en el desarrollo y finalización del proyecto. Todos los errores cometidos en las fases iníciales serán arrastrados por todas y cada una de las fases siguientes hasta que sean subsanados, lo cual resulta tanto más difícil cuanto más se haya avanzado en el proyecto.

Base de datos de Access

Una base de datos está compuesta de estructuras que almacenan información, así como de elementos que saben cómo manejar e interpretar esa información.

De este modo, una base de datos de Access está compuesta de:

* Tablas,
* Consultas sobre estas tablas,
* Formularios para presentar, editar y añadir información, además de para ofrecer un alto nivel de interacción con el usuario,
* Informes para presentar e imprimir resúmenes o documentos como facturas, informes de ventas, etc.
* Macros y código Access Basic para modificar y extender el comportamiento por defecto de todos los elementos anteriormente descritos.

Las aplicaciones de base de datos

Se dice que una base de datos es un conjunto de tablas, consultas, formularios, informes, macros y módulos. A una base de datos e le dice que es una aplicación cuando todos estos elementos se encuentran diseñados de forma interrelacionada, formando un todo. El añadirle la palabra “aplicación” es una especie de etiqueta de calidad.

Así, un usuario de una aplicación de base de datos no necesitará conocer el funcionamiento de Access ya que interactuará con los datos contenidos en nuestro sistema a través de un conjunto de formularios.

¿Para qué sirve la base de datos?

Sin duda la pregunta más importante al inicio del desarrollo es precisamente ésta ¿Qué se espera de la base de datos? ¿Qué necesidades tiene que cubrir?

Una base de datos se crea para dar solución a un problema o a un conjunto de ellos. Para ello, lo primero que debemos hacer es conocer el problema. Una vez conocido, deberíamos definirlo de forma detallada.

Para definir el problema con un nivel de detalle suficiente hay que preguntarse qué se espera de la base de datos en forma detallada: ¿Qué informes tiene que imprimir? ¿Cuáles serán sus formularios? ¿Qué búsquedas se desean? Si existe algún tipo de formulario impreso que hubiera que introducir en la base de datos, o que la base de datos tenga que imprimir, etc.

Después de definir el problema, ya se sabe de forma concreta cuáles son las necesidades que podría cubrir la base de datos. Por ello, en esta fase es en la que se van a analizar las soluciones que ofrece la base de datos.

Una vez que esta fase se considere finalizada, y se pase a la siguiente, no se debería añadir, eliminar o modificar la descripción de aquello que se espera de la base de datos. Por ello es muy importante quedar plenamente satisfecha con el resultado de esta fase.

El problema de la duplicación y eliminación de la información

En la siguiente figura aparece una tabla de productos y los datos de los proveedores que sirven dichos productos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Id | Nombre de producto | Nombre de compañía | Dirección |
| 1 | 1 | Té Charamsala | Exotic Liquids | 49 Gilbert St |
| 2 | 2 | Cerveza tibetana Barley | Exotic Liquids | 49 Gilbert St |
| 3 | 3 | Sirope de regaliz | Exotic Liquids | 49 Gilbert St |
| 4 | 4 | Especias Cajun de chef Anton | New Orleans Cajun Dielights | P.O.Box 78934 |
| 5 | 5 | Mezcla Gumbo de chef Anton | New Orleans Cajun Dielights | P.O.Box 78934 |
| 6 | 6 | Mermelada de grosellas de la abuela | Grandma Kally´s Homestead | 707 Oxford Rd. |
| 7 | 7 | Peras secas orgánicas del tio Bob | Grandma Kally´s Homestead | 707 Oxford Rd. |
| 8 | 8 | Salsa de arándanos Northwoods | Grandma Kally´s Homestead | 707 Oxford Rd. |
| 9 | 9 | Buey Mishi Kobe | Tokyo Tradero | 9-8 Sekimai |
| 10 | 10 | Pez espada | Tokyo Tradero | 9-8 Sekimai |
| 11 | 11 | Queso Cabrales | Cooperativa de quesos Las Cabras | Calle del Rosal 4 |
| 12 | 12 | Queso Manchego la Pastora | Cooperativa de quesos Las Cabras | Calle del Rosal 4 |
| 13 | 13 | Algas Kombu | Mayumi´s | 92 Setsuko |
| 14 | 14 | Cuajada de judías | Mayumi´s | 92 Setsuko |
| 15 | 15 | Salsa de soja baja en sodio | Mayumi´s | 92 Setsuko |
| 16 | 65 | Salsa de pimiento picante de Luisiana | New Orleans Cajun Dielights | P.O.Box 78934 |
| 17 | 66 | Especies picantes de Luisiana | New Orleans Cajun Dielights | P.O.Box 78934 |
| 18 | 74 | Queso de soja Longlife | Tokyo Tradero | 9-8 Sekimai |

Observe la siguiente figura

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Id | Nombre de producto | Nombre de compañía | Dirección |
| 1 | 1 | Té Charamsala | Exotic Liquids | 49 Gilbert St |
| 2 | 2 | Cerveza tibetana Barley | Exotic Liquids | 40 Gilbert St |
| 3 | 3 | Sirope de regaliz | Exotic Liquids | 49 Gilbert St |
| 4 | 4 | Especias Cajun de chef Anton | New Orleans Cajun Dielights | P.O.Box 78934 |
| 5 | 5 | Mezcla Gumbo de chef Anton | New Orleans Cajun Dielights | P.O.Box 78934 |
| 6 | 6 | Mermelada de grosellas de la abuela | Grandma Kally´s Homestead | 707 Oxford Rd. |
| 7 | 7 | Peras secas orgánicas del tio Bob | Grandma Kally´s Homestead | 707 Oxford Rd. |
| 8 | 8 | Salsa de arándanos Northwoods | Grandma Kally´s Homestead | 707 Oxford Rd. |
| 9 | 9 | Buey Mishi Kobe | Tokyo Tradero | 9-8 Sekimai |
| 10 | 10 | Pez espada | Tokyo Tradero | 9-8 Sekimai |
| 11 | 11 | Queso Cabrales | Cooperativa de quesos Las Cabras | Calle del Rosal 4 |
| 12 | 12 | Queso Manchego la Pastora | Cooperativa de quesos Las Cabras | Calle del Rosal 4 |
| 13 | 13 | Algas Kombu | Mayumi´s | 92 Setsuko |
| 14 | 14 | Cuajada de judías | Mayumi´s | 92 Setsuko |
| 15 | 15 | Salsa de soja baja en sodio | Mayumi´s | 92 Setsuko |
| 16 | 65 | Salsa de pimiento picante de Luisiana | New Orleans Cajun Dielights | P.O.Box 78934 |
| 17 | 66 | Especies picantes de Luisiana | New Orleans Cajun Dielights | P.O.Box 78934 |
| 18 | 74 | Queso de soja Longlife | Tokyo Tradero | 9-8 Sekimai |

En este caso aparece en el segundo registro de la tabla (segunda línea) una dirección mal escrita de proveedor que sirve el producto. Pero, ¿cómo sabe la máquina que precisamente esa dirección es la que está mal escrita?, ¿y si fueran los demás registros los que tuvieran mal la dirección y la del segundo registro fuera la correcta?, ¿y si aparecieran un número mucho más elevado de registros con varias direcciones diferentes para un mismo proveedor?

Imagine que tenemos en nuestra tabla un proveedor que únicamente sirve un producto. Al dar de alta el producto en nuestra base de datos tendremos también que introducir los datos del proveedor, ya que estos datos (los del proveedor) no existen en ningún otro producto de la base de datos. Si, por cualquier motivo, hay que eliminar este producto de la base de datos, con él también se perderán los datos del proveedor. Quizá nos hubiera interesado eliminar los datos del producto sin perder los datos del proveedor, pero el diseño d la base de datos no nos lo permite.

Lo ideal habría sido separar la información de los productos y la información de los proveedores, y tener alguna herramienta que nos asocie productos con proveedores. Precisamente para suplir estas y otras carencias surge el modelo relacional.

Modelo relacional

Access es un gestor de base de datos relacionales. Una base de datos relacional es aquella que además de los elementos presentes en las bases de datos (tablas, consultas, informes, formularios, etc.) incluye un modo de relacionar las tablas. Es por ello por lo que se denomina *Relaciones* a estas herramientas.

Siguiendo el ejemplo, para evitar los problemas expuestos anteriormente, se deben almacenar los productos en una tabla “Productos” y los proveedores, en una tabla “Proveedores”.

Si tenemos una tabla “Proveedores” y otra tabla “Productos”, lo ideal será que la base de datos “conozca” que tienen algo que ver, ya que los productos son vendidos por los proveedores. Así, “Productos” y “Proveedores” dejan de ser elementos independientes desde el punto de vista de la base de datos

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Id | Nombre de producto |  | Nombre de compañía | Dirección |
| 1 | 1 | Té Charamsala |  | Exotic Liquids | 49 Gilbert St |
| 2 | 2 | Cerveza tibetana Barley |  |  |  |
| 3 | 3 | Sirope de regaliz |  |  |  |
| 4 | 15 | Salsa de soja baja en sodio |  | Grandma Kally´s Homestead | 707 Oxford Rd. |
| 5 | 6 | Mermelada de grosellas de la abuela |  |  |  |
| 6 | 7 | Peras secas orgánicas del tio Bob |  |  |  |
| 7 | 8 | Salsa de arándanos Northwoods |  |  |  |
| 8 | 9 | Buey Mishi Kobe |  |  |  |
| 9 | 10 | Pez espada |  | Tokyo Tradero | 9-8 Sekimai |
| 10 | 11 | Queso Cabrales |  |  |  |
| 11 | 12 | Queso Manchego la Pastora |  | Cooperativa de quesos Las Cabras | Calle del Rosal 4 |
| 12 | 13 | Algas Kombu |  |  |  |
| 13 | 14 | Cuajada de judías |  | Mayumi´s | 92 Setsuko |
| 14 | 74 | Queso de soja Longlife |  |  |  |
| 15 | 5 | Mezcla Gumbo de chef Anton |  |  |  |
| 16 | 4 | Especias Cajun de chef Anton |  |  |  |
| 17 | 65 | Salsa de pimiento picante de Luisiana |  | New Orleans Cajun Dielights | P.O.Box 78934 |
| 18 | 66 | Especies picantes de Luisiana |  |  |  |

Ahora existe un único lugar donde se almacenan los datos de un producto y un único lugar donde se almacenan los datos de un proveedor (contrariamente a lo que ocurría antes)

Evidentemente, un planteamiento de este tipo exige un estudio previo a la implementación. Como venimos afirmando a lo largo del texto, no se puede simplemente comenzar con la implementación, sin haber dato una serie de pasos antes.

El diagrama Entidad – Relación

Para representar las tablas que existen en una base de datos, así como las relaciones que existen entre ellas, se utilizan los diagramas “entidad – relación”. En estos diagramas se muestran las tablas de la base de datos y las relacione que existen entre ellas

Elaboración del diagrama entidad – Relación

Para proceder con éxito a la elaboración del diagrama Entidad – Relación, debemos tener claros los siguientes conceptos:

* Una **entidad** es un elemento que existe (de forma tangible o intangible), y que puede distinguirse de otros elementos.
* Una **relación** crea algún tipo de asociación entre dos elementos (entidades) distintos.

En el ejemplo, cada producto es una entidad, así como cada proveedor es también otra entidad de distinto tipo.

Los conjuntos de entidad

Un conjunto de entidad no es más que una agrupación de entidades del mismo tipo. En nuestro caso “Proveedores” sería uno, al igual que “Productos” o “Pedidos”, para agregar uno nuevo.

Productos

Pedidos

Proveedores

Conjunto de relaciones

Conjunto de relaciones

Los conjuntos de relación

Los conjuntos de relación, de forma análoga, están formados por todas las relaciones existentes entre los conjuntos de entidad. Los conjuntos de relación son más difíciles de percibir que los conjuntos de entidad.

Parece claro que “Proveedores” es un conjunto de entidad, pero no lo es tanto que haya que crear un conjunto de relación entre “Proveedores” y “Pedidos”.

Determinación de las entidades del sistema

Las entidades son cosas o elementos distinguibles en el sistema que se pueden agrupar en tres grandes bloques:

* Físicas: objetos físicos del sistema como son: personas, elementos, nóminas, facturas
* Abstractas: entidades organizativas como son: proyectos, departamentos, presupuestos, etc.
* Sucesos: situaciones que ocurren o suceden como son: entregas, fallos en el sistema, ventas, cobros, etc.

Determinación de los conjuntos de entidad

En un mismo conjunto de entidad deben estar los elementos que tengan las mismas propiedades.

Por tanto, todos los productos estarían en el conjunto “Productos” y ninguno en el conjunto “Proveedores”

Determinación de los atributos

Todavía resta completar el diagrama entidad - relación añadiendo toda la información interesante para que, al estar documentado al máximo, sea lo más auto explicativo posible. De este modo, entre otras cosas, debemos determinar los atributos de cada conjunto y mostrarlos en el diagrama.

Los atributos son, por lo tanto, las propiedades que tienen los elementos que forman un conjunto. Se muestran los atributos al lado de cada conjunto de entidad. Además se debe tener en cuenta que uno de los atributos debe actuar como identificador de cada entidad del conjunto, por ejemplo un número de orden, número de documento, nombre (si no se repite), etc.

Un identificador es un atributo que distingue de forma unívoca a una entidad dentro del conjunto de entidades. Dentro de un mismo conjunto de entidades, no pueden existir dos entidades diferentes con el mismo identificador. Por ejemplo, un identificador de una persona en un censo de población podría ser un DNI.

Cada atributo de la entidad debe estar relacionado directamente con el tema de la entidad

Esto parece evidente, pero es una buena regla a tener en cuenta a la hora de determinar los atributos de un conjunto de entidad. Por ejemplo, si tenemos un conjunto entidad “Productos”, a primera vista no parece que el teléfono del proveedor que vende el producto sea un atributo adecuado para el conjunto entidad, no porque no esté relacionado, sino porque no lo está directamente.

Nunca incluya datos derivados o calculados

Por ejemplo, si tiene un conjunto de entidad “Pedidos”, no incluya el costo total del pedido si puede calcularlo a partir de los detalles del pedido ya almacenados.

Incluya toda la información necesaria

A pesar de que se debe cuidar el número de campos y su longitud para no malgastar espacio en el disco rígido, trate de definir todos los atributos necesarios. Si una información puede resultar útil, inclúyala. Es mejor definir un atributo ahora que más tarde.

Determinación del tipo de relaciones

Según el número de relaciones en las que puede aparecer involucrada una determinada entidad, estas pueden ser: **1:1** (uno a uno), **1:N** (uno a varios) y **N:N** (varios a varios).

Relaciones uno a uno

En esta relación se determina que cada entidad tan sólo puede estar asociada a otra. Por ejemplo, si en la base de datos de una empresa tenemos una entidad “Empleados” y otra entidad “Equipo de futbol de la empresa”, cada empleado aparecerá solo una vez en el conjunto de entidades “Equipo…”

Aparte de estos casos especiales, las relaciones uno a uno se suelen emplear para ampliar información acerca de una determinada entidad, información que no deseamos incluir entre los atributos de la misma entidad, sino que queremos almacenar de forma independiente. Antes de crear una relación uno a uno por esta causa, asegúrese de que lo que realmente quiere hacer no es agregar campos a la tabla.

Relaciones uno a varios

Es el tipo más usual de relación. Por ejemplo, si tenemos una entidad “Proveedores” y una entidad “Pedidos” y cada pedido se hace a un solo proveedor, cada proveedor podrá tener asociados varios pedidos, pero un pedido tan solo podrá tener asociado un proveedor.

Relaciones varios a varios

Este tipo de relación es un poco más complejo. Imaginemos ahora que llevamos un control de productos en nuestra base de datos. Los proveedores venden productos, de modo que cada proveedor vende varios productos. Pero, a su vez, un mismo producto puede ser vendido por varios proveedores. En este caso tendríamos que usar una relación varios a varios.

Del esquema Entidad – Relación a Access

Las entidades

Cada conjunto entidades se transforma inmediatamente en una tabla. El conjunto de entidades “Pedidos” se transforma en la tabla “Pedidos”. Sus atributos pasan a ser sus campos y su identificador pasa a ser su clave primaria o principal.

Las relaciones

Para establecer relaciones entre las tablas hay que tener en cuenta ciertas características que deben tener dichas tablas

Uno a varios

Para este caso para la tabla que tiene cardinalidad 1, el atributo identificador será la clave principal. Este mismo atributo se repetirá en la tabla de cardinalidad varios (clave externa9, aunque en este caso no será un atributo identificador, ni tampoco será campo clave, dejando esta función para otro campo que identifique unívocamente los registros (entidades) de la tabla. Por último tanto en la tabla de cardinalidad 1 como en la de cardinalidad varios este campo en común deberá tener el mismo tipo de dato.

Uno a uno

En la relación uno a uno el atributo identificador será la clave principal en ambas tablas

Varios a varios

Para las relaciones varios a varios no basta con agregar una clave externa, sino que necesitamos una tabla auxiliar. Situaremos esta tabla entre cada relación de las entidades relacionadas “varios a varios” de forma que crearemos dos conjuntos de relación “uno a varios”