

## 2006 HSCC Programming Problem: Donate for Disaster Relief

Over the past year, thousands of people across the United States have suffered from the effects of hurricanes, floods, fires, tornadoes and other natural disasters. Those more fortunate have contributed time and money to help people in need, but they often have to choose between charitable organizations, making each donation separately.

Your assignment is to create a "Donate for Disaster Relief" application where someone can choose from one or more charities and make a donation, all from one web site. Charities can also use the site to see how much money has been raised for their organization.

Donors (people who want to give money), will have two choices from the home page of the web site. Returning donors (those who have donated before) will be able to login with a userid and password and see their profile. New donors will need to create a profile before they can start donating to charities. Once successfully logged into the application, a donor can choose one or more charities and enter a dollar amount to contribute. The application will keep track of which charities the donor has picked, the dollars for each charity, and the total dollars for all donations made during this session on the web site. Once the donor has made all their selections, they can "check out" of the application and pay for their donations using the credit card stored in their profile. At any time, the donor can change their charity selections, start over from the beginning, or abandon the session by logging off the web site.

Charities (organizations who receive donations) may login to the application using their own userid and password to see a report which shows total contributions to date along with a list of all donors and the amount each donor has contributed to that charity.

Both donors and charities (type of users) will login to the application using the same home page:

- **Returning donor.** Donors who have already created a profile should enter a userid and password that matches an entry on the Donor table. The password should be protected when entered. If the userid and password are correct, the donor will then go to a page where they see a list of all their donations. The application should delete any rows from the Session and Session\_Choices tables.
- **New donor.** New donors should click a link that takes them to a page where they can create a profile for themselves. The application should delete any rows from the Session and Session\_Choices tables.
- **Charity.** A user for a charity should enter a userid and password that matches an entry on the Charity table. The password should be protected when entered. If the userid and password are correct, the application should take them to a page that displays a donation report.

Once logged in, donors will be able to do the following things:

- **Display their donations.** The donor will see a page showing their first and last name, address (street, city, state, and zip code), preferred charity name (with logo graphic) if one has been chosen, and total amount of all donations made to date. After this information, the page will display a list of all donations made by the donor, displaying the date of the donation, the charity name (with logo graphic), and the donation amount. The list should be sorted by date, from most recent to oldest.

From here, the donor may edit their profile, make a donation, or exit the application.

- **Create/edit their profile.** The donor will be able to enter new information or edit current information in their profile. New donors should start with all fields empty; returning donors will see their profile information. (See table descriptions for a list of all fields and formats.) The donor **must** enter first name, last name, street address, city, and zip code. They **must** select a credit card type (from a defined list), enter a credit card number, and the expiration date of the credit card (month and year, both from a defined list). They have the **option** to choose a preferred charity from a list of charities on the Charity table. Once all information has been entered or updated, the donor clicks a button to save the information. *choose ID & Pass*

From here, the donor may display their donations, make a donation, or exit the application.

- September
  - October
  - November
  - December
- The credit card expiration year list is:
  - 2006
  - 2007
  - 2008
  - 2009
- The application uses the concept of a “session” to keep track of the donor’s choices while they are making donations. The Session and Session\_Choices tables are designed for this purpose. Once the donor has confirmed their choices during checkout, the Session\_Choices information should be written to the Donations table. Be careful to make sure that you delete entries from the Session and Session\_Choices tables properly, so that the donor always sees their most recent choices, but that the choices are not lost if the donor decides to go to another page before checkout.

The application must provide for flexible navigation and ease of use. Messages, prompts, navigation tips, proper controls, and display of code descriptions will all be important to consider as part of your design. Your team will need to decide what pages are needed, and the overall application should have a consistent look and feel to the user.

- **Make a donation.** The donor will be able to make donations to one or more charities from the list of charities on the Charity table. The page will display the donor first and last name, address (street, city, state, and zip code), and preferred charity name (with logo graphic) if one has been chosen, and a running total of the donations made during this visit to the web site (called a session). For each charity, display the charity name (with logo graphic), and a field for the donation amount. The list of charities should be sorted by charity name. If a preferred charity has been chosen by the donor, that charity should appear first in the list. Your team will need to determine the best way for the donor to select which charities will receive a donation during the session.

You will need to use the Session and Session\_Choices tables to keep track of the choices that the donor makes.

Once the donor has made all their choices and entered the dollar amount for each selected charity, they can click a "review donation" button that will update the running total. If they are happy with the answer, they can click another button to let them checkout.

At any time, the donor can decide to go edit their profile or display their donations, so the application must keep track of what the donor has entered so far, and display it again when they return to the "make a donation" page during that session. The donor may also choose to exit the application without completing their donations.

- **Checkout.** The donor will be able to review all their choices, and then confirm that they really want to make the donations. The checkout page should display the same information as the "make a donation" page (just described) as well as the donor credit card number and expiration date. The credit card number display should be protected, except for the last 4 digits (for example, \*\*\*\*\*1234).

The donor will click a button to confirm their donations, which should display an acknowledgement message, and the application should then permanently record the information. The donor should also be able to change their mind and cancel the donations. In either case, once checkout is finished, the donor should be sent back to the page that displays their donations.

Once logged in, a charity user will be able to do the following things:

- **Display a donation report.** The user from a charity will be able to see a page that displays the name of the charity (and logo graphic) and the total amount of all donations made to that charity. Following that information will be a list of each donation made to that charity, displaying the date of the donation, the name of the donor, the address of the donor, and the dollar amount of the donation. The list should be sorted by date (most recent to oldest), then by donor last name.

The user from the charity should be able to click a button to print this page, or exit the application.

Please note the following:

- Where something is identified by a code, but has a text description, display the description.
- The credit card type list is:
  - American Express
  - Discover
  - Mastercard
  - Visa
- The credit card expiration date month list is:
  - January
  - February
  - March
  - April
  - May
  - June
  - July
  - August

## **Specific Programming Information**

### *File Locations*

All supplied files for the problem can be found in: **c:\HSCC2006**; other development files are in the directories previously provided to you.

### *Database*

The database and tables (with test data), have been supplied to you as a MySQL database named **hsc2006**.

Reference the attached documents for specific field names, data types, keys, and a relationship diagram.

While changing the database structure (types of fields and relationships) is possible, it is not suggested.

### *Graphics*

All graphics are not embedded in the database, but supplied as a URL to the location of the graphic file on your team's hard drive in following directory. You must use the specific graphics related to certain database records, but you may use any of the other graphics as you choose in the application.

**c:\HSCC2006\Images**

### *Code Values*

You must display the value associated with a code (e.g., codes like CharityID).

```

DROP TABLE IF EXISTS `hsc2006`.`donor`;
CREATE TABLE `hsc2006`.`donor` (
  `DonorID` char(10) NOT NULL default '', COMMENT 'Unique identifier for a donor',
  `DonorLastName` varchar(45) NOT NULL default '', COMMENT 'Last name of donor',
  `DonorFirstName` varchar(45) NOT NULL default '', COMMENT 'First name of donor',
  `DonorAddress` varchar(45) NOT NULL default '', COMMENT 'Street address of donor',
  `DonorCity` varchar(45) NOT NULL default '', COMMENT 'City of donor',
  `DonorState` char(2) NOT NULL default '', COMMENT 'State abbreviation for donor',
  `DonorZip` char(9) NOT NULL default '', COMMENT 'zip code/Postal code for donor',
  `DonorPassword` varchar(45) NOT NULL default '', COMMENT 'Password for donor to use application',
  `DonorTotalDonations` decimal(9,2) NOT NULL default '0.00' COMMENT 'Total of all donations made by donor',
  `DonorCreditCard` char(16) NOT NULL default '', COMMENT 'Credit card number of donor',
  `DonorCCName` varchar(45) NOT NULL default '', COMMENT 'Credit card name for donor',
  `DonorCEXpmth` varchar(45) NOT NULL default '', COMMENT 'Credit card expiration month',
  `DonorCEXpyr` char(4) NOT NULL default '', COMMENT 'Credit card expiration year',
  `DonorPreferredCharity` char(10) default NULL COMMENT 'ID of preferred charity from Charity table',
  PRIMARY KEY (`DonorID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Contains a record for each donor';

DROP TABLE IF EXISTS `hsc2006`.`charity`;
CREATE TABLE `hsc2006`.`charity` (
  `CharityID` char(10) NOT NULL default '', COMMENT 'Unique identifier of a charity',
  `CharityName` varchar(45) NOT NULL default '', COMMENT 'Name of charity',
  `CharityLogoURL` varchar(45) NOT NULL default '', COMMENT 'URL link to charity logo graphic',
  `CharityPassword` varchar(45) NOT NULL default '', COMMENT 'Password for charity to use application',
  PRIMARY KEY (`CharityID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Contains a record for each charity';

DROP TABLE IF EXISTS `hsc2006`.`donations`;
CREATE TABLE `hsc2006`.`donations` (
  `DonorID` char(10) NOT NULL default '', COMMENT 'Donor ID from Donor table',
  `CharityID` char(10) NOT NULL default '', COMMENT 'Charity ID from Charity table',
  `DonationDateTime` datetime NOT NULL default '0000-00-00 00:00:00' COMMENT 'Timestamp of donation',
  `DonationAmt` decimal(9,2) NOT NULL default '0.00' COMMENT 'Amount of donation',
  PRIMARY KEY (`DonorID`,`CharityID`,`DonationDateTime`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Contains donations made by a donor to a charity';

DROP TABLE IF EXISTS `hsc2006`.`session`;
CREATE TABLE `hsc2006`.`session` (
  `SessionID` datetime NOT NULL default '0000-00-00 00:00:00' COMMENT 'Timestamp for the session',
  `DonorID` char(10) NOT NULL default '', COMMENT 'Donor ID from the Donor table',
  PRIMARY KEY (`SessionID`,`DonorID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Contains the basic session information for a donor choices';

DROP TABLE IF EXISTS `hsc2006`.`session_choices`;
CREATE TABLE `hsc2006`.`session_choices` (
  `SessionID` datetime NOT NULL default '0000-00-00 00:00:00' COMMENT 'Session ID from Session table',
  `SessionID` datetime NOT NULL default '0000-00-00 00:00:00' COMMENT 'Session ID from Session table',

```

```
`DonorID` char(10) NOT NULL default '' COMMENT 'Donor ID from Session table',
`CharityID` char(10) NOT NULL default '' COMMENT 'Charity ID from Charity table',
`Choiceamt` decimal(9,2) NOT NULL default '0.00' COMMENT 'Donation amount for this choice',
PRIMARY KEY (`SessionID`,`DonorID`,`CharityID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Contains each donation choice a donor has made during the session';

LOAD DATA INFILE 'c:/HSCC2006/Donor.csv' INTO TABLE donor fields terminated by ',' ENCLOSED BY '' ESCAPED BY '\\'
LINES TERMINATED BY '\r\n';
LOAD DATA INFILE 'c:/HSCC2006/charity.csv' INTO TABLE charity fields terminated by ',' ENCLOSED BY '' ESCAPED BY
'' LINES TERMINATED BY '\r\n';
LOAD DATA INFILE 'c:/HSCC2006/Donations.csv' INTO TABLE donations fields terminated by ',' ENCLOSED BY '' ESCAPED BY
'' LINES TERMINATED BY '\r\n';
```

## Relationships for DisasterRelief

Sunday, July 30, 2006

