Script SQL

--

-- PostgreSQL database dump

--

--

-- SCHEMA: GESTbook

--

DROP SCHEMA IF EXISTS GESTbook CASCADE;

CREATE SCHEMA GESTbook;

SET client\_encoding = 'UTF8';

SET search\_path = news, pg\_catalog;

--

-- DOMAIN: status\_type

--

CREATE DOMAIN status\_type AS text NOT NULL DEFAULT 'Editada'::text

CONSTRAINT status\_type\_check CHECK ((VALUE = ANY (ARRAY['Editada'

::text, 'Publicada'::text, 'Arquivada'::text])));

--

-- TABLE: funcionário

--

CREATE TABLE funcionarios (

user\_id INTEGER PRIMARY KEY,

nome CHARACTER VARYING(128) NOT NULL,

categoria CHARACTER VARYING(64) NOT NULL,

telefone CHARACTER VARYING(24) NOT NULL UNIQUE,

nif CHARACTER VARYING(9) NOT NULL UNIQUE,

);

ALTER TABLE ONLY funcionarios

COMMENT ON TABLE funcionarios IS 'Informacao acerca dos funcionarios';

COMMENT ON COLUMN funcionarios.user\_id IS 'Numero de Identificacao;

COMMENT ON COLUMN funcionarios.nome IS 'Nome';

COMMENT ON COLUMN funcionarios.telefone IS 'número de Telefone';

COMMENT ON COLUMN funcionarios.nif IS 'nif';

--

-- TABLE: Centro Custos

--

CREATE TABLE CentroCustos(

nid INTEGER PRIMARY KEY,

tipocusto CHARACTER VARYING(64) NOT NULL,

);

ALTER TABLE ONLY CentroCustos

ADD CONSTRAINT news\_pkey PRIMARY KEY (nid);

ALTER TABLE ONLY news

ADD CONSTRAINT news\_nauthor\_fkey FOREIGN KEY (nauthor) REFERENCES authors(aid) ON UPDATE SET NULL ON DELETE SET NULL;

COMMENT ON TABLE CentroCustos IS 'Informacao acerca dos pedidos cabimentação';

COMMENT ON COLUMN CentroCustos.nid IS 'Numero de identificacao do centro';

COMMENT ON COLUMN CentroCustos.tipocusto IS 'Indicação se o centro de custos em causa foi criado para eventos, projetos, estudantes ou para custos pessoais';

--

-- TABLE: cabimentação

--

CREATE TABLE cabimentacao (

numero\_id INTEGER PRIMARY KEY,

estado CHARACTER VARYING(30) NOT NULL,

descritivo CHARACTER VARYING(32) NOT NULL,

valordeorçamento INTEGER NOT NULL,

);

ALTER TABLE ONLY cabimentacao

COMMENT ON TABLE cabimentacao IS 'informaticao a cerca dos orçamentos pedidos pelos funcionários';

COMMENT ON COLUMN cabimentacao.numero\_id IS 'username';

COMMENT ON COLUMN cabimentacao .estado IS 'Estado';

COMMENT ON COLUMN cabimentacao.descritivo IS 'Descrição' ;

--

-- TABLE: Operação

--

CREATE TABLE operacao(

numero\_id INTEGER PRIMARY KEY,

data date NOT NULL,

tipomovimento CHARACTER VARYING(64) NOT NULL,

tiposuporte CHARACTER VARYING(64) NOT NULL,

descritivo CHARACTER VARYING(64) NOT NULL,

montante INTEGER;

beneficiante CHARACTER VARYING(128) NOT NULL,

temporeembolso INTEGER

);

ALTER TABLE ONLY operacao

ADD CONSTRAINT news\_pkey PRIMARY KEY (nid);

ALTER TABLE ONLY news

ADD CONSTRAINT news\_nauthor\_fkey FOREIGN KEY (nauthor) REFERENCES authors(aid) ON UPDATE SET NULL ON DELETE SET NULL;

COMMENT ON TABLE operacao IS 'Informacao acerca as operações envolvidas por uma cabimentaçao';

COMMENT ON COLUMN CentroCustos.nid IS 'Numero de identificacao do centro';

COMMENT ON COLUMN CentroCustos.tipomovimento IS 'tipo de movimento';

COMMENT ON COLUMN CentroCustos.tiposuporte IS 'Tipo de Suporte';

COMMENT ON COLUMN CentroCustos.descitivo IS 'Descrição';

COMMENT ON COLUMN CentroCustos.montante IS 'Montante';

COMMENT ON COLUMN CentroCustos.beneficiante IS 'Benefeciante';

COMMENT ON COLUMN CentroCustos.temporeembolso IS 'Tempo de Reembolso';

COMMENT ON COLUMN CentroCustos.data IS 'Data efectuada';

--

-- INDEX: ndtate\_idx

CREATE INDEX ndtate\_idx ON news USING btree (ndate);

--

-- Data for authors

--

INSERT INTO authors (aname, aemail, aphone, aage) VALUES ('Donald Knuth', 'donald@stan.edu', '12.234.5678', 67);

INSERT INTO authors (aname, aemail, aphone, aage) VALUES ('Alan Kay', 'alan@xerox.com', '34.567.8901', 64);

INSERT INTO authors (aname, aemail, aphone, aage) VALUES ('Steve Jobs', 'steve@apple.com', '23.456.7890', 54);

--

-- Data for news

--

INSERT INTO news (ntitle, nbody, ndate, nauthor, nstatus) VALUES ('Literate Programming', 'Literate programming is a methodology that combines a programming language with a documentation language, thereby making programs more robust, more portable, more easily maintained, and arguably more fun to write than programs that are written only in a high-level language. The main idea is to treat a program as a piece of literature, addressed to human beings rather than to a computer. The program is also viewed as a hypertext document, rather like the World Wide Web. (Indeed, I used the word WEB for this purpose long before CERN grabbed it!) This book is an anthology of essays including my early papers on related topics such as structured programming, as well as the article in The Computer Journal that launched Literate Programming itself. The articles have been revised, extended, and brought up to date.', '2011-11-12', 1, 'Publicada');

INSERT INTO news (ntitle, nbody, ndate, nauthor, nstatus) VALUES ('The Art of Computer Programming', 'At the end of 1999, these books were named among the best twelve physical-science monographs of the century by American Scientist, along with: Dirac on quantum mechanics, Einstein on relativity, Mandelbrot on fractals, Pauling on the chemical bond, Russell and Whitehead on foundations of mathematics, von Neumann and Morgenstern on game theory, Wiener on cybernetics, Woodward and Hoffmann on orbital symmetry, Feynman on quantum electrodynamics, Smith on the search for structure, and Einstein''s collected papers. Wow! ', '2011-11-11', 1, 'Publicada');

INSERT INTO news (ntitle, nbody, ndate, nauthor, nstatus) VALUES ('The TeXbook', 'Volumes B and D contain the source code for TeX and METAFONT, respectively, written with the literate programming methodology. ``Never before has a computer program of this size been spelled out so clearly and completely.'''' Volume E contains precise definitions of about 500 letters, numerals, and other symbols, all described with METAFONT. This method of description produces an essentially infinite variety of well-hinted fonts from a single collection of specifications. Readers learn how to make their own personal variations, simply by changing a few parameters. Special symbols that are needed for unusual applications can also be created by using the many examples in this book as a model. The details captured in these METAFONT programs reveal many previously unpublished tricks that type designers have been learning during the past centuries. ', '2011-11-11', 2, 'Publicada');

INSERT INTO news (ntitle, nbody, ndate, nauthor, nstatus) VALUES ('The METAFONTbook ', 'The first 26 printings of The TeXbook and the first 9 printings of The METAFONTbook each contained dozens of revisions, based on feedback from readers; but the present printings are ``stable'''' and need not be changed further. Very few errors of any significance are known in the current editions. Those that have been reported can be found in the Comprehensive TeX Archive Network (CTAN), a large collection of freely available material about TeX and METAFONT that appears in dozens of mirror sites through the world. The official errata lists appear in subdirectory systems/knuth/errata. File errata.tex in that subdirectory lists the errors in the current printings; other files errata.one, errata.two, ..., errata.ten give information about older printings. Files tex82.bug, mf84.bug, and cm85.bug contain records of all past changes to TeX, METAFONT, and Computer Modern. (A detailed history of all errors in TeX can be found in chapters 10 and 11 of the book Literate Programming and in chapter 34 of the book Digital Typography.) The 20th printing of The TeXbook and the 6th printing of TeX: The Program, in April 2000, incorporate corrections to all errors known to exist in those books at that time.', '2011-11-12', 3, 'Publicada');

INSERT INTO news (ntitle, nbody, ndate, nauthor, nstatus) VALUES ('Things a Computer Scientist Rarely Talks About ', 'After an introductory first session, the second lecture focuses on the interaction of randomization and religion, since randomization has become a key area of scientific interest during the past few decades. The third lecture considers questions of language translation, with many examples drawn from the author''s experiments in which random verses of the Bible were analyzed in depth. The fourth one deals with art and aesthetics; it illustrates several ways in which beautiful presentations can greatly deepen our perception of difficult concepts. The fifth lecture discusses what the author learned from the "3:16 project," a personal exploration of Biblical literature which he regards as a turning point in his own life. ', '2011-11-13', 1, 'Editada');

INSERT INTO news (ntitle, nbody, ndate, nauthor, nstatus) VALUES ('Exemplo de submissão do RA', 'Por forma a assegurar que não irão existir problemas derivados de redundância de informação devida a dependências funcionais, é verificado se a Base de Dados está em conformidade com a forma normal de Boyce-Codd (BCNF). A partir do conhecimento do domínio do problema e tomando um esquema de relação de cada vez, são identificadas as dependências funcionais que aí se verificam. Para a relação estar na BCNF é necessário e suficiente que só existam dependências de conjuntos de atributos que contenham uma chave [RG02]. ', '2011-11-11', 1, 'Editada');

INSERT INTO news (ntitle, nbody, ndate, nauthor, nstatus) VALUES ('What is Smarty?', 'Smarty is a template engine for PHP. More specifically, it facilitates a manageable way to separate application logic and content from its presentation. This is best described in a situation where the application programmer and the template designer play different roles, or in most cases are not the same person. ', '2011-10-10', 3, 'Arquivada');

--

-- Data for users; Type: TABLE DATA; Schema: news; Owner: devel

--

INSERT INTO users (uuser, upass, uname) VALUES ('admin', '698dc19d489c4e4db73e28a713eab07b', 'Administrador');

INSERT INTO users (uuser, upass, uname) VALUES ('gates', '1992544155f5192b6959a136745f7281', 'William Gates II');

INSERT INTO users (uuser, upass, uname) VALUES ('mariana', '698dc19d489c4e4db73e28a713eab07b', 'Mariana do Teste');

-- end