University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project

Milestone #: 2

Date: March 1, 2024

Group Number: 82

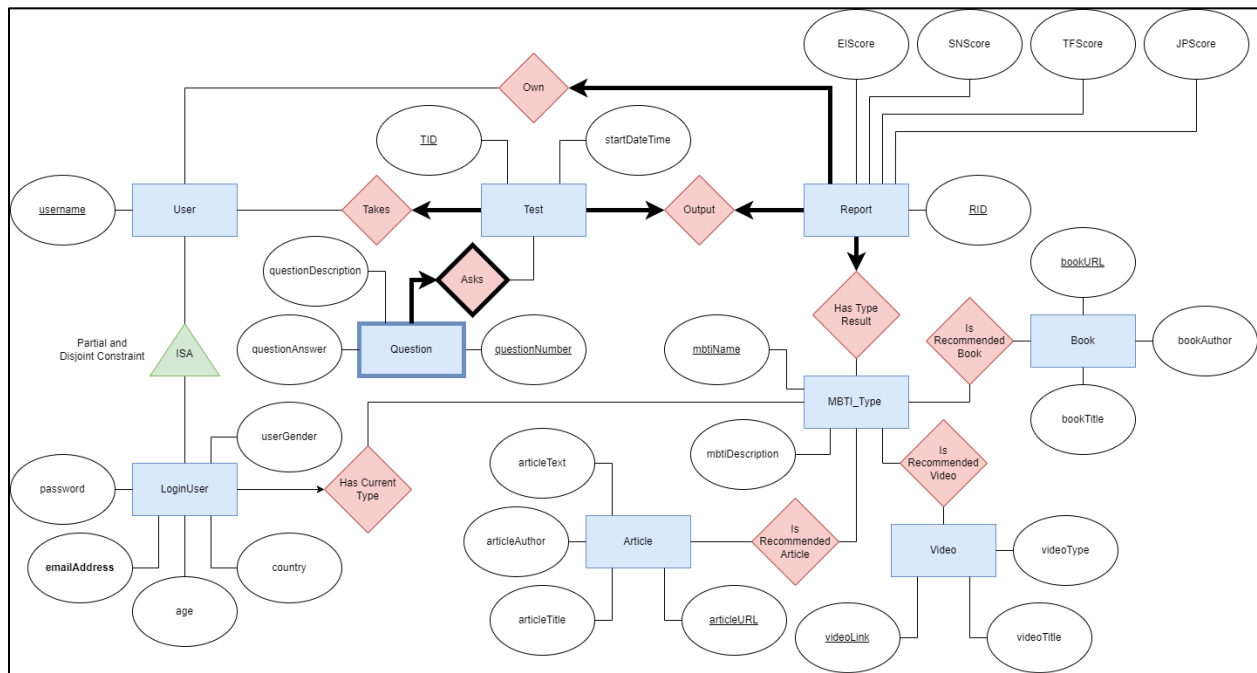| Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Herman Liao	16267817	j2s2z	hl01234@student.ubc.ca
William Wang	14671341	gox2s	williamwang921220@gmail.com
Yujie Chen	60392396	x4j6p	yujiec@student.ubc.ca

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Project Description:

The relational database management system that we will create is a personality test website. This database management system would be in the domain of psychology, as it analyzes users' personalities and teaches users about them. The users may take the tests to learn more about themselves to become more aware of their own behaviours and thoughts, as well as learn about others' behaviours and thoughts.

ER Diagram:

Relational Schema:

Notation: <u>Primary Key</u>, **Candidate Key**, *Foreign Key*

Entities:

User (<u>username</u>: CHAR)
- Primary Key: username
- None of the attributes can be null

LoginUser(<u>*username*</u>: CHAR, **emailAddress**: CHAR, *mbtiName*: INT, password: CHAR, age INT, country VARCHAR, userGender CHAR)
- Primary Key: username
- Candidate Key: emailAddress
- Foreign Key: username, mbtiName
- username, password, and emailAddress are not null

Question(<u>questionNumber</u>: INT, <u>*TID*</u>: INT, questionAnswer: INT, questionDescription: TEXT)
- Primary Key: questionNumber + TID
- Foreign Key: TID
- None of the attributes can be null
- Question is a weak entity belongs to Test

MBTI_Type(**mbtiName**: CHAR, mbtiDescription: TEXT)
- Primary Key: mbtiName
- None of the attributes can be null

Book(<u>bookURL</u>: VARCHAR, bookTitle: TEXT, bookAuthor: TEXT)
- Primary Key: bookURL
- None of the attributes can be null

Article(<u>articleURL</u>: VARCHAR, articleTitle: TEXT, articleAuthor: TEXT, articleText: TEXT)
- Primary Key: articleURL
- None of the attributes can be null

Video(<u>videoLink</u>: VARCHAR, videoTitle: VARCHAR, videoType: VARCHAR)
- Primary Key: videoLink
- videoLink and videoTitle can not be null

Relationships:

Outputs(<u>TID</u>: INT, **RID**: INT, **startDateTime**: DATETIME, *username*: CHAR, *mbtiName*: CHAR, EIScore: REAL, SNScore: REAL, TFScore: REAL, JPScore: REAL)
- Primary Key: TID
- Candidate Key: RID, startTime + username
- Foreign Key: username, mbtiName
- None of the attributes can be null
- Test and Report have a one-to-one total participation relationship with each other, so their tables have been merged into the Outputs table here
- Report is in a many-to-one relationship with User in the Own relationship, with Report having total participation
- Test is in a many-to-one relationship with User in the Takes relationship, with Test having total participation
- Report is in a many-to-one relationship with MBTI_Type in the HasTypeResult relationship, with Report having total participation

IsRecommendedBook(<u>*bookURL*</u>: VARCHAR, <u>*mbtiName*</u>: CHAR)
- Primary Key: bookURL, mbtiName
- Foreign Key: bookURL, mbtiName
- None of the attributes can be null
- Because Book and MBTI_Type are in a many-to-many relationship, this table has been created to represent it

IsRecommendedArticle(<u>*articleURL*</u>: VARCHAR, <u>*mbtiName*</u>: CHAR)
- Primary Key: articleURL, mbtiName
- Foreign Key: articleURL, mbtiName
- None of the attributes can be null
- Because Article and MBTI_Type are in a many-to-many relationship, this table has been created to represent it

IsRecommendedVideo(<u>*videoLink*</u>: VARCHAR, <u>*mbtiName*</u>: CHAR)
- Primary Key: videoLink, mbtiName
- Foreign Key: videoLink, mbtiName
- None of the attributes can be null
- Because Video and MBTI_Type are in a many-to-many relationship, this table has been created to represent it

Functional Dependencies:

User: No functional dependencies

LoginUser:
username → emailAddress, mbtiName, password, age, country, userGender
emailAddress → username, mbtiName, password, age, country, userGender

Question:
questionNumber, TID → questionAnswer, questionDescription

MBTI_Type:
mbtiName → mbtiDescription

Book:
bookURL → bookTitle, bookAuthor

Article:
articleURL → articleTitle, articleAuthor, articleText
articleTitle, articleAuthor → articleText

Video:
videoLink → videoTitle, videoType

Outputs:
TID → startDateTime, username, RID
username, startDateTime → TID, RID
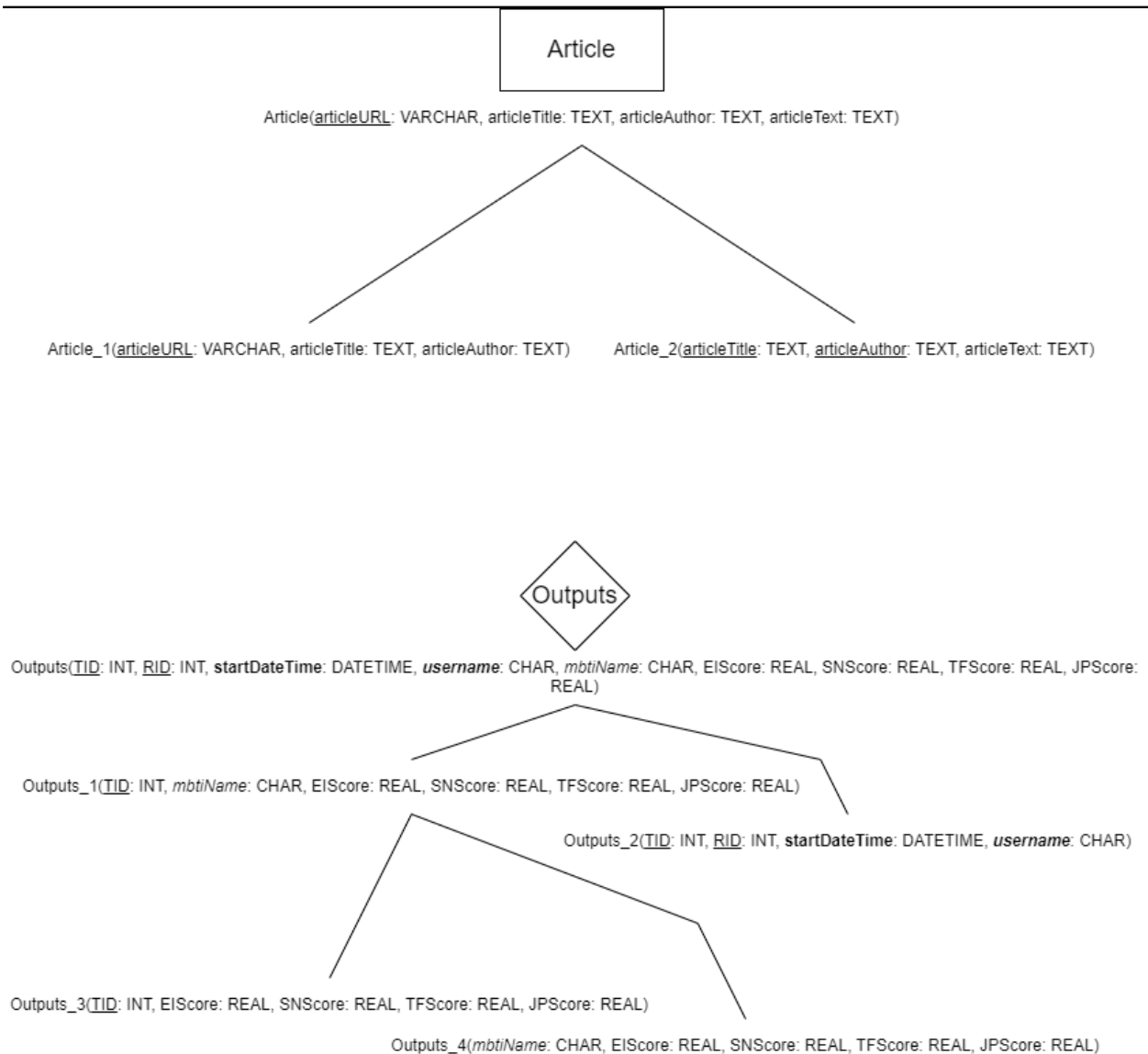RID → EIScore, SNScore, TFScore, JPScore, mbtiName, username, TID
EIScore, SNScore, TFScore, JPScore → mbtiName

IsRecommendedBook: No functional dependencies

IsRecommendedArticle: No functional dependencies

IsRecommendedVideo: No functional dependencies

Normalization:

Article

Article(articleURL: VARCHAR, articleTitle: TEXT, articleAuthor: TEXT, articleText: TEXT)

Article_1(articleURL: VARCHAR, articleTitle: TEXT, articleAuthor: TEXT)

Article_2(articleTitle: TEXT, articleAuthor: TEXT, articleText: TEXT)

Outputs

Outputs(TID: INT, RID: INT, **startDateTime**: DATETIME, *username*: CHAR, *mbtiName*: CHAR, EIScore: REAL, SNScore: REAL, TFScore: REAL, JPScore: REAL)

Outputs_1(TID: INT, *mbtiName*: CHAR, EIScore: REAL, SNScore: REAL, TFScore: REAL, JPScore: REAL)

Outputs_2(TID: INT, RID: INT, **startDateTime**: DATETIME, *username*: CHAR)

Outputs_3(TID: INT, EIScore: REAL, SNScore: REAL, TFScore: REAL, JPScore: REAL)

Outputs_4(*mbtiName*: CHAR, EIScore: REAL, SNScore: REAL, TFScore: REAL, JPScore: REAL)

SQL DDL Statements:

AUTO_INCREMENT affects the addition of tuples into the relational database. Instead of the value for the attribute being specified, each insertion will give the attribute a unique value, then increment the value that will be given to the next insertion. The first insertion gives a value of 1, then the next insertion gives a value of 2, etc.

```sql
CREATE TABLE User (
    username INT NOT NULL AUTO_INCREMENT,
    PRIMARY KEY (username),
);

CREATE TABLE LoginUser (
    username INT,
    mbtiName CHAR(4),
    username VARCHAR(255) NOT NULL,
    password VARCHAR(255) NOT NULL,
    emailAddress VARCHAR(255) NOT NULL,
    age INT,
    country VARCHAR(255),
    userGender CHAR(1),
    PRIMARY KEY (username),
    UNIQUE (emailAddress),
    FOREIGN KEY (username) REFERENCES User(username) ON DELETE
    CASCADE
    FOREIGN KEY (mbtiName) REFERENCES MBTI_Type(mbtiName) ON
    DELETE CASCADE
);

CREATE TABLE Test (
    TID INT NOT NULL AUTO_INCREMENT,
    RID INT NOT NULL,
    username INT NOT NULL,
    startTime DATETIME NOT NULL,
    testGender CHAR(1),
    PRIMARY KEY (TID),
    UNIQUE (startTime, username),
    FOREIGN KEY (username) REFERENCES User(username) ON DELETE
CASCADE
    FOREIGN KEY (RID) REFERENCES Report_2(RID) ON DELETE CASCADE
);

CREATE TABLE Question (
    questionNumber INT NOT NULL AUTO_INCREMENT,
    TID INT NOT NULL,
    questionDescription TEXT NOT NULL,
    questionAnswer INT NOT NULL,
```

```
    PRIMARY KEY (questionNumber, TID),
    FOREIGN KEY (TID) REFERENCES Test(TID) ON DELETE CASCADE
);

CREATE TABLE Outputs_3 (
    TID INT NOT NULL,
    EIScore REAL,
    SNScore REAL,
    TFScore REAL,
    JPScore REAL,
    mbtiName INT NOT NULL,
    PRIMARY KEY (EIScore, EIScore, EIScore, EIScore),
    FOREIGN KEY (EIScore, EIScore, EIScore, EIScore) REFERENCES
    Outputs_4(EIScore, EIScore, EIScore, EIScore) ON DELETE
    CASCADE
);
CREATE TABLE Outputs_3 (
    TID INT NOT NULL,
    EIScore REAL,
    SNScore REAL,
    TFScore REAL,
    JPScore REAL,
    mbtiName INT NOT NULL,
    PRIMARY KEY (EIScore, EIScore, EIScore, EIScore),
    FOREIGN KEY (EIScore, EIScore, EIScore, EIScore) REFERENCES
    Outputs_4(EIScore, EIScore, EIScore, EIScore) ON DELETE
    CASCADE
);


CREATE TABLE Report_2 (
    RID INT NOT NULL AUTO_INCREMENT,
    TID INT NOT NULL,
    username INT NOT NULL,
    mbtiName INT NOT NULL,
    EIScore REAL NOT NULL,
    SNScore REAL NOT NULL,
    TFScore REAL NOT NULL,
    JPScore REAL NOT NULL,
    reportGender CHAR(1) NOT NULL,
    PRIMARY KEY (RID),
    FOREIGN KEY (TID) REFERENCES Test(TID) ON DELETE CASCADE,
    FOREIGN KEY (EIScore) REFERENCES Report_1(EIScore) ON DELETE
    CASCADE,
    FOREIGN KEY (SNScore) REFERENCES Report_1(SNScore) ON DELETE
    CASCADE,
```

```
      FOREIGN KEY (TFScore) REFERENCES Report_1(TFScore) ON DELETE
      CASCADE,
      FOREIGN KEY (JPScore) REFERENCES Report_1(JPScore) ON DELETE
      CASCADE,
      FOREIGN KEY (reportGender) REFERENCES Report_1(reportGender)
      ON DELETE CASCADE
);

CREATE TABLE MBTI_Type_1 (
     mbtiName CHAR(4),
     typeDescription TEXT,
     PRIMARY KEY (mbtiName)
);

CREATE TABLE MBTI_Type_2 (
     mbtiName INT NOT NULL AUTO_INCREMENT,
     mbtiName CHAR(4) NOT NULL,
     mbtiImage IMAGE,
     mbtiGender CHAR(1) NOT NULL,
     UNIQUE(mbtiImage)
     UNIQUE(mbtiName,mbtiGender)
     FOREIGN KEY (mbtiName) REFERENCES MBTI_Type_1(mbtiName) ON
     DELETE CASCADE
);


CREATE TABLE Book (
     BID INT AUTO_INCREMENT,
     bookURL VARCHAR(255),
     bookTitle TEXT NOT NULL,
     bookAuthor TEXT NOT NULL,
     PRIMARY KEY (bookURL),
     UNIQUE(bookURL)
);


CREATE TABLE Article_1 (
      articleTitle TEXT,
      articleAuthor TEXT,
      articleText TEXT,
      PRIMARY KEY (articleTitle, articleAuthor),
);

CREATE TABLE Article_2 (
      AID INT NOT NULL AUTO_INCREMENT,
      articleURL VARCHAR(255),
```

```
    articleTitle TEXT,
    articleAuthor TEXT,
    PRIMARY KEY (AID),
    UNIQUE (articleURL),
    FOREIGN KEY (articleTitle) REFERENCES
    Article_1(articleTitle) ON DELETE CASCADE,
    FOREIGN KEY (articleAuthor) REFERENCES
    Article_2(articleAuthor) ON DELETE CASCADE
);

CREATE TABLE Video (
    VID INT NOT NULL AUTO_INCREMENT,
    videoLink VARCHAR(255),
    videoType VARCHAR(255) NOT NULL,
    videoTitle VARCHAR(255) NOT NULL,
    PRIMARY KEY (VID)
);

CREATE TABLE IsRecommendedVideo (
    mbtiName INT,
    VID INT,
    PRIMARY KEY (mbtiName, VID),
    FOREIGN KEY (mbtiName) REFERENCES MBTI_Type_2(mbtiName) ON
    DELETE CASCADE,
    FOREIGN KEY (VID) REFERENCES Video(VID) ON DELETE CASCADE
);

CREATE TABLE IsRecommendedBook (
    mbtiName INT,
    BID INT,
    PRIMARY KEY (mbtiName, BID),
    FOREIGN KEY (mbtiName) REFERENCES MBTI_Type_2(mbtiName) ON
    DELETE CASCADE,
    FOREIGN KEY (BID) REFERENCES Book(BID) ON DELETE CASCADE
);

CREATE TABLE IsRecommendedArticle (
    mbtiName INT,
    AID INT,
    PRIMARY KEY (mbtiName, AID),
    FOREIGN KEY (mbtiName) REFERENCES MBTI_Type_2(mbtiName) ON
    DELETE CASCADE,
    FOREIGN KEY (AID) REFERENCES Article(AID) ON DELETE CASCADE
);
```

SQL INSERT Statements:

```
INSERT INTO User ()
VALUES (), (), (), (), (), (), (), (), (), ();



INSERT INTO LoginUser (username, mbtiName, password,
emailAddress, age, country)
VALUES
("1234567", "", "EEEE", "hl01234@student.ubc.ca", 13, "Canada"),
("realDon", "INTJ", "pr3$IdenT2024",
"donaldtrump@whitehouse.com", 77, "USA"),
("mrB3ast", "", "raidShadowLegends", "beast@gmail.com", 40,
"Australia"),
("xxxDART", "", "Emp!re$1ThL0rD", "darthvader@hotmail.com", 436,
"North Korea"),
("margare", "", "1ForrEST1!", "margThatch@outlook.com", 97,
"United Kingdom");

INSERT INTO Question (questionNumber, TID, questionDescription,
questionAnswer)
VALUES
(1, 1, "I often get tired when talking with others.", 5),
(2, 1, "I base my decisons off of facts and logic.", 1),
(3, 1, "I enjoy trying new things and gaining new experiences.",
7),
(4, 1, "I am optimistic about life.", 4),
(5, 1, "When working with others, I often agree with their
ideas.", 2);

INSERT INTO Book (bookTitle, bookAuthor, bookURL)
VALUES
("The fine art of small talk", "Debra FineDebra Fine",
"https://www.amazon.ca/gp/product/0349436177/ref=ox_sc_saved_tit
le_1?smid=A3HMVO0974R9WW&psc=1"),
("Fourth Wing", "Rebecca Yarros", "https://www.amazon.ca/Fourth-
Wing-Rebecca-
Yarros/dp/1649374046/ref=sr_1_17?crid=J08O0JUYD2X&dib=eyJ2IjoiMS
J9.vNEzwUzblDG9DYS6PZUzfoZSEfqWC486A0b3uu6mmobAglriinvog2_zxDfT7
9DctCXKXN4nnAvv412l3zlrru7Nu10n5l-Jzrt6-
aEJ4idI0WQqFNT0RbOmWIzIudjMrPjtOA0PK9Vb-y7mQKrk3qftL5LOiE_um-
_4nFP75WG-
BMFPwfVsyJIjf8jjNA5jx_JExjhb7NYsffb3Y0QngCPlQZ5u9akj0JN_XpNlXIDe
_rBbMswc4L6rsx88HofwgS2Z7l9ALQm4SeP3fWqLiyaYVtY3i1pONInhPZ_m2dw.
HPyK5lakUgtVrCRj0LTpR1_zN8L675duhlzBE_JniJ0&dib_tag=se&keywords=
books&qid=1708911065&sprefix=book%2Caps%2C177&sr=8-17"),
```

```
("A Good Girl's Guide to Murder", "Holly Jackson",
"https://www.amazon.ca/Good-Girls-Guide-
Murder/dp/1984896393/ref=sr_1_25?crid=J08O0JUYD2X&dib=eyJ2IjoiMS
J9.vNEzwUzblDG9DYS6PZUzfoZSEfqWC486A0b3uu6mmobAglriinvog2_zxDfT7
9DctCXKXN4nnAvv412l3zlrru7Nu10n5l-Jzrt6-
aEJ4idI0WQqFNT0RbOmWIzIudjMrPjtOA0PK9Vb-y7mQKrk3qftL5LOiE_um-
_4nFP75WG-
BMFPwfVsyJIjf8jjNA5jx_JExjhb7NYsffb3Y0QngCPlQZ5u9akj0JN_XpNlXIDe
_rBbMswc4L6rsx88HofwgS2Z7l9ALQm4SeP3fWqLiyaYVtY3i1pONInhPZ_m2dw.
HPyK5lakUgtVrCRj0LTpR1_zN8L675duhlzBE_JniJ0&dib_tag=se&keywords=
books&qid=1708911065&sprefix=book%2Caps%2C177&sr=8-25"),
("The Lightning Thief", "Rick Riordan",
"https://www.amazon.ca/Percy-Jackson-Olympians-Lightning-
Thief/dp/0786838655/ref=sr_1_1?crid=3Q0CD01M4BUFW&dib=eyJ2IjoiMS
J9.CM5vyHYrhOAiS1wynN6IqReO61-51L8pL05MwoaYU2alDoSunEDRhm61tzv-
q-
oTsqgeOw6UDSrzVMat8tRhVtBkPlGHZ6qU2TKt0RFrj3wsyWRcwMHArGoWFd81c-
YhOSYhgFqkvZC9bpqPRkdCvt7HI3-UKeDeVMgHS1cHktQAE6F7pT0oMSowoO1-
FSi6pwOoFGu7I47axITQuix3MGyL32T0qsPPS-J2UhzeSj6p8-
OyjD4DS4YiMEvWguwNUOfqMjk0BO5bvalhLJMd3gRx0mfH4t9bIAKVv_v7mr4.oA
B3ZphmKxvKVqqHfH7ID8vFTGOiRqhatJYPhQ_iv5g&dib_tag=se&keywords=th
e+lightning+thief&qid=1708911236&sprefix=the+lightnin%2Caps%2C16
3&sr=8-1");
("Atomic Habits", "James Clear", "https://www.amazon.ca/Atomic-
Habits-Proven-Build-
Break/dp/0735211299/ref=sr_1_1?crid=11WLFTSNIX22U&dib=eyJ2IjoiMS
J9.SN-
noe4EpN9PPXpEWtjSQCeagM7OcCP6C52WJbT5ej14hcboatudTPVwZVxiTWMQ786
BKwLISTG40M9jrP5N-cq4qIrj1IKH-
8znuCHL1YYUVnS0xH6N_PmfmoQPWzyj2csQaLiu2UtwnPB9OVWcEyhgX4Rd_1fs7
i4eUcq4nHm2WvrWAf_YSDBagL0uAfUQJ1oTzrQHVW07Ej81L8B8D666c3hJzLIy5
cPHF0mexrE.SfPK9LI1dM5SShMbGCTqQYe2osP_ktW-fWkU-
FORukQ&dib_tag=se&keywords=atomic+habits&qid=1708911339&s=books&
sprefix=atomic+%2Cstripbooks%2C207&sr=1-1");


INSERT INTO Video (videoType, videoTitle, videoLink)
VALUES
("Short YouTube Video", "The quiet power of introverts | BBC
Ideas", "https://www.youtube.com/watch?v=o1Y4Z0oh1GE"),
("Movie", "In the Forests of Siberia",
"https://www.imdb.com/title/tt4643432/"),
("Animation
Movie","Suzume","https://www.imdb.com/title/tt16428256/?ref_=nv_
sr_srsg_0_tt_6_nm_2_q_suzume"),
("Documentary","Becoming Warren
Buffett","https://www.imdb.com/title/tt6438096/"),
```

```
("Tiktok Short", "This man gave a baby goat berries and got a
bonus",
"https://www.tiktok.com/@bored_meerkatyt/video/71618367183358558
74?lang=en");

INSERT INTO Article_2 (articleURL, articleTitle, articleAuthor)
VALUES
("https://www.16personalities.com/articles/personality-types-vs-
stereotypes", "Personality Types vs. Stereotypes", "KYLE"),
("https://www.16personalities.com/articles/self-expression-
boundaries-and-love-consuls-esfj", "Self-Expression, Boundaries,
and Love: Consuls (ESFJ)", "TRACI"),
("https://www.16personalities.com/articles/opposites-attract-
when-advocates-infj-and-entrepreneurs-estp-fall-in-love",
"Opposites Attract: When Advocates (INFJ) and Entrepreneurs
(ESTP) Fall in Love", "DARRELL"),
("https://www.16personalities.com/articles/from-
miscommunication-to-effective-team-communication-insights-into-
every-personality-type", "From Miscommunication to Effective
Team Communication: Insights into Every Personality Type",
"CARLY"),
("https://www.16personalities.com/articles/future-proof-your-
career-upskilling-advice-for-every-personality-type", "Future-
Proof Your Career: Upskilling Advice for Every Personality
Type", "TRACI");

INSERT INTO Article_1 (articleTitle, articleAuthor, ArticleText)
VALUES
("Personality Types vs. Stereotypes", "KYLE", "The 16
personality types are unique, with each group defined by their
traits and the behavioral tendencies that go along with them.
Based on statistics, personality typing allows for accurate
descriptions and assessments of people. But there's a difference
between personality typing, which can be very helpful in many
areas of life, and stereotyping, which can be harmful.
Understanding that difference is important for the sake of
fairness, accuracy, and usefulness. Let's consider how
personality types differ from stereotypes."),
("Self-Expression, Boundaries, and Love: Consuls (ESFJ)",
"TRACI", "Self-expression might seem easier for some people than
for others. By all outward appearances, Consuls (ESFJ), with
their Extraverted and Feeling traits, tend to fall into this
camp. People with this personality type are among the most open
about who they are and what they think, generally presenting
themselves as "open books."),
```

```
("Opposites Attract: When Advocates (INFJ) and Entrepreneurs
(ESTP) Fall in Love", "DARRELL", "Have you ever seen a couple
and asked yourself, "What do they see in each other?" Why two
people fall in love can be an incredibly mysterious thing. But
the fact is, there are no hard-and-fast rules on who attracts
whom and why. It can be very shortsighted to assume that a
relationship will fail based on how many characteristics a
couple shares or doesn't share. How two people come together
when they have little in common can be baffling, but no
combination of personality types should ever be counted out in
romance."),
("From Miscommunication to Effective Team Communication:
Insights into Every Personality Type", "CARLY", "Ask any
successful team leader why effective team communication is
important, and they'll likely have an extensive list of reasons
at the ready to report back. A great leader knows that good
communication is quite possibly the single most important
quality of high-performing teams."),
("Future-Proof Your Career: Upskilling Advice for Every
Personality Type", "TRACI", "I've been thinking a lot about
upskilling lately. Ever since the launch of ChatGPT and other
generative AI tools, I have been constantly reminded of just how
important it is to be proactive about staying up-to-date on the
latest advances in technology and future-proofing your
career.");
```