users,"CREATE TABLE `users` (

`user\_id` varchar(20) NOT NULL,

`username` varchar(50) NOT NULL,

`password` varchar(255) NOT NULL,

`role` enum('superadmin','admin','teacher','student','guest') NOT NULL,

`staff\_id` varchar(20) DEFAULT NULL COMMENT '职工号',

`student\_id` varchar(20) DEFAULT NULL COMMENT '学号',

`login\_count` int DEFAULT '0',

`name` varchar(50) DEFAULT NULL,

`gender` enum('male','female') DEFAULT NULL,

`department` varchar(100) DEFAULT NULL,

`major` varchar(100) DEFAULT NULL COMMENT '学生专业',

`title` varchar(100) DEFAULT NULL COMMENT '教师职称',

`email` varchar(100) DEFAULT NULL,

`phone` varchar(20) DEFAULT NULL,

`introduction` text,

`updated\_at` timestamp NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

`status` enum('active','inactive','locked') DEFAULT 'active',

`last\_login\_time` timestamp NULL DEFAULT NULL,

`last\_login\_ip` varchar(50) DEFAULT NULL,

`avatar\_url` varchar(255) DEFAULT NULL,

`created\_at` timestamp NULL DEFAULT CURRENT\_TIMESTAMP,

PRIMARY KEY (`user\_id`),

UNIQUE KEY `idx\_staff\_id` (`staff\_id`),

UNIQUE KEY `idx\_student\_id` (`student\_id`),

KEY `idx\_role` (`role`),

KEY `idx\_username\_status` (`username`,`status`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci"

translation\_reviews,"CREATE TABLE `translation\_reviews` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`file\_url` varchar(255) DEFAULT NULL COMMENT '文件URL',

`file\_size` bigint DEFAULT NULL COMMENT '文件大小(字节)',

`submit\_time` datetime DEFAULT NULL COMMENT '提交时间',

`version` int NOT NULL DEFAULT '1' COMMENT '版本号',

`language\_accuracy\_score` decimal(5,2) DEFAULT NULL COMMENT '语言准确性分数',

`content\_integrity\_score` decimal(5,2) DEFAULT NULL COMMENT '内容完整性分数',

`format\_score` decimal(5,2) DEFAULT NULL COMMENT '格式规范分数',

`total\_score` decimal(5,2) DEFAULT NULL COMMENT '总分',

`language\_accuracy\_comment` text COMMENT '语言准确性评语',

`content\_integrity\_comment` text COMMENT '内容完整性评语',

`format\_comment` text COMMENT '格式规范评语',

`general\_comment` text COMMENT '总体评语',

`improvement\_suggestions` text COMMENT '改进建议',

`status` enum('pending','reviewing','revision\_needed','completed') DEFAULT 'pending',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

KEY `idx\_student\_status` (`student\_id`,`status`),

CONSTRAINT `translation\_reviews\_ibfk\_1` FOREIGN KEY (`student\_id`) REFERENCES `students` (`student\_id`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='外文翻译评审表'"

topics,"CREATE TABLE `topics` (

`topic\_id` int NOT NULL AUTO\_INCREMENT,

`title` varchar(255) NOT NULL,

`type` varchar(50) NOT NULL,

`source` varchar(100) DEFAULT NULL,

`teacher\_id` varchar(20) NOT NULL,

`description` text,

`major\_requirements` text,

`student\_requirements` text,

`status` enum('draft','pending','approved','rejected','selected') DEFAULT 'draft',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`topic\_id`),

KEY `idx\_teacher\_status` (`teacher\_id`,`status`),

KEY `idx\_teacher\_id` (`teacher\_id`),

CONSTRAINT `topics\_ibfk\_1` FOREIGN KEY (`teacher\_id`) REFERENCES `users` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=328 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci"

thesis\_submissions,"CREATE TABLE `thesis\_submissions` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL COMMENT '学生ID',

`advisor\_id` varchar(20) DEFAULT NULL COMMENT '指导教师ID',

`version` int NOT NULL DEFAULT '1' COMMENT '提交版本',

`file\_url` varchar(255) NOT NULL COMMENT '论文文件URL',

`file\_size` int DEFAULT NULL COMMENT '文件大小(KB)',

`title` varchar(255) NOT NULL COMMENT '论文标题',

`keywords` varchar(255) DEFAULT NULL COMMENT '关键词',

`abstract` text COMMENT '摘要',

`status` enum('submitted','advisor\_reviewing','advisor\_rejected','advisor\_approved','review\_assigned','reviewing','completed','revision\_needed') DEFAULT 'submitted' COMMENT '状态',

`advisor\_review\_status` enum('pending','reviewing','rejected','approved') DEFAULT 'pending' COMMENT '导师审核状态',

`advisor\_review\_time` datetime DEFAULT NULL COMMENT '导师审核时间',

`advisor\_review\_comments` text COMMENT '导师审核意见',

`advisor\_approval\_time` datetime DEFAULT NULL COMMENT '导师确认时间',

`advisor\_comments` text COMMENT '导师意见',

`submit\_time` datetime DEFAULT CURRENT\_TIMESTAMP COMMENT '提交时间',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

KEY `student\_id` (`student\_id`),

KEY `advisor\_id` (`advisor\_id`),

CONSTRAINT `thesis\_submissions\_ibfk\_1` FOREIGN KEY (`student\_id`) REFERENCES `students` (`student\_id`) ON DELETE CASCADE,

CONSTRAINT `thesis\_submissions\_ibfk\_2` FOREIGN KEY (`advisor\_id`) REFERENCES `users` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=11 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='论文提交记录表'"

thesis\_reviews\_assignment,"CREATE TABLE `thesis\_reviews\_assignment` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL COMMENT '学生ID',

`reviewer\_id` varchar(20) NOT NULL COMMENT '评阅人ID',

`review\_type` enum('advisor','peer') NOT NULL COMMENT '评阅类型：导师评阅/同行评阅',

`status` enum('pending','completed') DEFAULT 'pending' COMMENT '评阅状态',

`assigned\_at` datetime DEFAULT CURRENT\_TIMESTAMP COMMENT '分配时间',

`completed\_at` datetime DEFAULT NULL COMMENT '完成时间',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

KEY `student\_id` (`student\_id`),

KEY `reviewer\_id` (`reviewer\_id`),

CONSTRAINT `thesis\_reviews\_assignment\_ibfk\_1` FOREIGN KEY (`student\_id`) REFERENCES `students` (`student\_id`) ON DELETE CASCADE,

CONSTRAINT `thesis\_reviews\_assignment\_ibfk\_2` FOREIGN KEY (`reviewer\_id`) REFERENCES `users` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='论文评阅分配表'"

thesis\_review\_details,"CREATE TABLE `thesis\_review\_details` (

`id` int NOT NULL AUTO\_INCREMENT,

`assignment\_id` int NOT NULL COMMENT '关联thesis\_reviews\_assignment表的ID',

`score\_innovation` decimal(5,2) DEFAULT NULL COMMENT '创新性评分',

`score\_quality` decimal(5,2) DEFAULT NULL COMMENT '质量评分',

`score\_workload` decimal(5,2) DEFAULT NULL COMMENT '工作量评分',

`score\_writing` decimal(5,2) DEFAULT NULL COMMENT '写作水平评分',

`total\_score` decimal(5,2) DEFAULT NULL COMMENT '总评分',

`review\_comments` text COMMENT '评阅意见',

`review\_time` datetime DEFAULT NULL COMMENT '评阅时间',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

KEY `assignment\_id` (`assignment\_id`),

CONSTRAINT `thesis\_review\_details\_ibfk\_1` FOREIGN KEY (`assignment\_id`) REFERENCES `thesis\_reviews\_assignment` (`id`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='论文评阅详情表'"

task\_assignments,"CREATE TABLE `task\_assignments` (

`task\_id` int NOT NULL AUTO\_INCREMENT,

`teacher\_id` varchar(20) NOT NULL,

`student\_id` varchar(20) NOT NULL,

`topic\_id` int DEFAULT NULL,

`selection\_id` int DEFAULT NULL,

`title` varchar(200) NOT NULL,

`content` text NOT NULL,

`requirements` text,

`deadline` datetime DEFAULT NULL,

`status` enum('draft','published','completed') DEFAULT 'draft',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`task\_id`),

KEY `teacher\_id` (`teacher\_id`),

KEY `student\_id` (`student\_id`),

KEY `topic\_id` (`topic\_id`),

KEY `selection\_id` (`selection\_id`),

CONSTRAINT `task\_assignments\_ibfk\_1` FOREIGN KEY (`teacher\_id`) REFERENCES `users` (`user\_id`),

CONSTRAINT `task\_assignments\_ibfk\_2` FOREIGN KEY (`student\_id`) REFERENCES `users` (`user\_id`),

CONSTRAINT `task\_assignments\_ibfk\_3` FOREIGN KEY (`topic\_id`) REFERENCES `topics` (`topic\_id`),

CONSTRAINT `task\_assignments\_ibfk\_4` FOREIGN KEY (`selection\_id`) REFERENCES `topic\_selections` (`selection\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=29 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci"

system\_messages,"CREATE TABLE `system\_messages` (

`message\_id` int NOT NULL AUTO\_INCREMENT,

`title` varchar(100) NOT NULL,

`content` text NOT NULL,

`type` enum('notice','warning','error') DEFAULT 'notice',

`status` enum('active','archived') DEFAULT 'active',

`created\_at` timestamp NULL DEFAULT CURRENT\_TIMESTAMP,

`created\_by` varchar(20) DEFAULT NULL,

`priority` enum('low','medium','high') DEFAULT 'medium',

`expires\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`message\_id`),

KEY `created\_by` (`created\_by`),

KEY `idx\_message\_status\_priority` (`status`,`priority`),

CONSTRAINT `system\_messages\_ibfk\_1` FOREIGN KEY (`created\_by`) REFERENCES `users` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=14 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci"

system\_logs,"CREATE TABLE `system\_logs` (

`log\_id` int NOT NULL AUTO\_INCREMENT,

`user\_id` varchar(20) DEFAULT NULL,

`action` varchar(100) NOT NULL,

`ip\_address` varchar(50) DEFAULT NULL,

`details` text,

`created\_at` timestamp NULL DEFAULT CURRENT\_TIMESTAMP,

PRIMARY KEY (`log\_id`),

KEY `idx\_user\_action\_time` (`user\_id`,`action`,`created\_at`),

CONSTRAINT `system\_logs\_ibfk\_1` FOREIGN KEY (`user\_id`) REFERENCES `users` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=1123 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci"

supervisors,"CREATE TABLE `supervisors` (

`supervisor\_id` varchar(20) NOT NULL,

`name` varchar(50) NOT NULL DEFAULT '未命名',

`title` varchar(50) DEFAULT NULL,

`department` varchar(100) DEFAULT NULL,

`phone` varchar(20) DEFAULT NULL,

`email` varchar(100) DEFAULT NULL,

`research\_area` text,

`max\_students` int DEFAULT '8',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`supervisor\_id`),

KEY `idx\_department` (`department`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci"

students,"CREATE TABLE `students` (

`student\_id` varchar(50) NOT NULL,

`name` varchar(50) NOT NULL,

`gender` enum('男','女') DEFAULT NULL,

`major` varchar(100) DEFAULT NULL,

`class\_name` varchar(50) DEFAULT NULL,

`phone` varchar(20) DEFAULT NULL,

`email` varchar(100) DEFAULT NULL,

`notes` text,

PRIMARY KEY (`student\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci"

student\_stage\_scores,"CREATE TABLE `student\_stage\_scores` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`task\_book\_score` decimal(5,2) DEFAULT NULL COMMENT '任务书分数',

`literature\_score` decimal(5,2) DEFAULT NULL COMMENT '文献综述分数',

`proposal\_score` decimal(5,2) DEFAULT NULL COMMENT '开题报告分数',

`translation\_score` decimal(5,2) DEFAULT NULL COMMENT '外文翻译分数',

`midterm\_score` decimal(5,2) DEFAULT NULL COMMENT '中期检查分数',

`thesis\_submit\_score` decimal(5,2) DEFAULT NULL COMMENT '论文提交分数',

`advisor\_review\_score` decimal(5,2) DEFAULT NULL COMMENT '导师评阅分数',

`peer\_review\_score` decimal(5,2) DEFAULT NULL COMMENT '同行评阅分数',

`defense\_score` decimal(5,2) DEFAULT NULL COMMENT '答辩分数',

`total\_weighted\_score` decimal(5,2) DEFAULT NULL COMMENT '总加权分数',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

KEY `student\_id` (`student\_id`),

CONSTRAINT `student\_stage\_scores\_ibfk\_1` FOREIGN KEY (`student\_id`) REFERENCES `students` (`student\_id`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='学生阶段分数表'"

proposal\_reviews,"CREATE TABLE `proposal\_reviews` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`teacher\_id` varchar(50) NOT NULL,

`version` int NOT NULL DEFAULT '1' COMMENT '版本号',

`file\_url` varchar(255) DEFAULT NULL COMMENT '开题报告文件路径',

`status` enum('not\_started','in\_progress','reviewing','revision\_needed','completed') DEFAULT 'not\_started' COMMENT '状态',

`research\_background\_score` decimal(5,2) DEFAULT NULL COMMENT '研究背景分数',

`technical\_route\_score` decimal(5,2) DEFAULT NULL COMMENT '技术路线分数',

`feasibility\_score` decimal(5,2) DEFAULT NULL COMMENT '可行性分析分数',

`innovation\_score` decimal(5,2) DEFAULT NULL COMMENT '创新点分数',

`total\_score` decimal(5,2) DEFAULT NULL COMMENT '总分',

`research\_background\_comment` text COMMENT '研究背景评语',

`technical\_route\_comment` text COMMENT '技术路线评语',

`feasibility\_comment` text COMMENT '可行性分析评语',

`innovation\_comment` text COMMENT '创新点评语',

`general\_comment` text COMMENT '总体评语',

`improvement\_suggestions` text COMMENT '改进建议',

`submit\_time` datetime DEFAULT NULL COMMENT '提交时间',

`review\_time` datetime DEFAULT NULL COMMENT '评审时间',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

KEY `student\_id` (`student\_id`),

KEY `teacher\_id` (`teacher\_id`),

CONSTRAINT `proposal\_reviews\_ibfk\_1` FOREIGN KEY (`student\_id`) REFERENCES `students` (`student\_id`) ON DELETE CASCADE,

CONSTRAINT `proposal\_reviews\_ibfk\_2` FOREIGN KEY (`teacher\_id`) REFERENCES `users` (`user\_id`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='开题报告评审表'"

progress\_stages,"CREATE TABLE `progress\_stages` (

`id` int NOT NULL AUTO\_INCREMENT,

`name` varchar(100) NOT NULL COMMENT '阶段名称',

`type` enum('task\_book','literature','proposal','translation','midterm','thesis\_submit','advisor\_review','peer\_review','defense') NOT NULL COMMENT '阶段类型',

`sequence` int NOT NULL COMMENT '阶段顺序',

`required\_files` tinyint(1) DEFAULT '1' COMMENT '是否需要提交文件',

`requires\_review` tinyint(1) DEFAULT '1' COMMENT '是否需要审核',

`weight` decimal(5,2) NOT NULL DEFAULT '0.00' COMMENT '分数权重',

`description` text COMMENT '阶段说明',

`status` enum('active','inactive') DEFAULT 'active' COMMENT '阶段状态',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `type` (`type`),

UNIQUE KEY `sequence` (`sequence`)

) ENGINE=InnoDB AUTO\_INCREMENT=36 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='论文阶段定义表'"

progress\_stage\_queue,"CREATE TABLE `progress\_stage\_queue` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`next\_stage\_type` enum('task\_book','literature','proposal','translation','midterm','thesis\_submit','advisor\_review','peer\_review','defense') NOT NULL,

`created\_at` timestamp NULL DEFAULT CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=24 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci"

progress\_records,"CREATE TABLE `progress\_records` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`stage\_type` enum('task\_book','literature','proposal','translation','midterm','thesis\_submit','advisor\_review','peer\_review','defense') NOT NULL COMMENT '阶段类型',

`status` enum('not\_started','in\_progress','reviewing','revision\_needed','completed','overdue') DEFAULT 'not\_started' COMMENT '状态',

`file\_url` varchar(255) DEFAULT NULL COMMENT '提交的文件URL',

`teacher\_comment` text COMMENT '教师评语',

`score` decimal(5,2) DEFAULT NULL COMMENT '分数',

`submit\_time` datetime DEFAULT NULL COMMENT '提交时间',

`review\_time` datetime DEFAULT NULL COMMENT '审核时间',

`deadline` datetime DEFAULT NULL COMMENT '截止时间',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `unique\_student\_stage` (`student\_id`,`stage\_type`),

CONSTRAINT `progress\_records\_ibfk\_1` FOREIGN KEY (`student\_id`) REFERENCES `students` (`student\_id`) ON DELETE CASCADE

) ENGINE=InnoDB AUTO\_INCREMENT=4445 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='学生进度记录表'"

next\_stage\_queue,"CREATE TABLE `next\_stage\_queue` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`current\_stage\_type` varchar(50) NOT NULL,

`next\_stage\_type` varchar(50) NOT NULL,

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`processed` tinyint(1) DEFAULT '0',

PRIMARY KEY (`id`),

KEY `processed` (`processed`)

) ENGINE=InnoDB AUTO\_INCREMENT=12 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci"

midterm\_reports,"CREATE TABLE `midterm\_reports` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`file\_url` varchar(255) DEFAULT NULL COMMENT '文件URL',

`file\_size` bigint DEFAULT NULL COMMENT '文件大小(字节)',

`submit\_time` datetime DEFAULT NULL COMMENT '提交时间',

`version` int NOT NULL DEFAULT '1' COMMENT '版本号',

`research\_progress\_score` decimal(5,2) DEFAULT NULL CO

MMENT '研究进度分数',

`technical\_ability\_score` decimal(5,2) DEFAULT NULL COMMENT '技术掌握分数',

`work\_attitude\_score` decimal(5,2) DEFAULT NULL COMMENT '工作态度分数',

`total\_score` decimal(5,2) DEFAULT NULL COMMENT '总分',

`progress\_comment` text COMMENT '进度评语',

`technical\_comment` text COMMENT '技术评语',

`attitude\_comment` text COMMENT '态度评语',

`improvement\_suggestions` text COMMENT '改进建议',

`status` enum('pending','reviewing','revision\_needed','completed') DEFAULT 'pending',

`teacher\_id` varchar(20) DEFAULT NULL COMMENT '评审教师ID',

`review\_time` datetime DEFAULT NULL COMMENT '评审时间',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

KEY `idx\_student\_status` (`student\_id`,`status`),

KEY `idx\_teacher\_id` (`teacher\_id`),

CONSTRAINT `midterm\_reports\_ibfk\_1` FOREIGN KEY (`student\_id`) REFERENCES `students` (`student\_id`) ON DELETE CASCADE,

CONSTRAINT `midterm\_reports\_ibfk\_2` FOREIGN KEY (`teacher\_id`) REFERENCES `users` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='中期报告表'"

literature\_reviews,"CREATE TABLE `literature\_reviews` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`teacher\_id` varchar(20) NOT NULL,

`file\_url` varchar(255) DEFAULT NULL COMMENT '文献综述文件路径',

`status` enum('not\_started','in\_progress','reviewing','revision\_needed','completed') DEFAULT 'not\_started',

`submit\_time` datetime DEFAULT NULL COMMENT '提交时间',

`review\_time` datetime DEFAULT NULL COMMENT '评阅时间',

`content\_score` decimal(5,2) DEFAULT NULL COMMENT '内容完整性分数',

`analysis\_score` decimal(5,2) DEFAULT NULL COMMENT '分析深度分数',

`structure\_score` decimal(5,2) DEFAULT NULL COMMENT '结构逻辑分数',

`writing\_score` decimal(5,2) DEFAULT NULL COMMENT '写作规范分数',

`total\_score` decimal(5,2) DEFAULT NULL COMMENT '总分',

`content\_comment` text COMMENT '内容评价',

`improvement\_suggestions` text COMMENT '改进建议',

`general\_comment` text COMMENT '总体评语',

`version` int DEFAULT '1' COMMENT '提交版本',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `unique\_student\_version` (`student\_id`,`version`),

KEY `idx\_teacher` (`teacher\_id`),

KEY `idx\_status` (`status`),

CONSTRAINT `fk\_literature\_student` FOREIGN KEY (`student\_id`) REFERENCES `students` (`student\_id`) ON DELETE CASCADE,

CONSTRAINT `fk\_literature\_teacher` FOREIGN KEY (`teacher\_id`) REFERENCES `users` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=7 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='文献综述评阅表'"

defense\_scores,"CREATE TABLE `defense\_scores` (

`id` int NOT NULL AUTO\_INCREMENT,

`arrangement\_id` int NOT NULL,

`teacher\_id` varchar(20) NOT NULL,

`presentation\_score` decimal(5,2) DEFAULT NULL COMMENT '答辩陈述分数',

`qa\_score` decimal(5,2) DEFAULT NULL COMMENT '问答表现分数',

`innovation\_score` decimal(5,2) DEFAULT NULL COMMENT '创新性分数',

`completion\_score` decimal(5,2) DEFAULT NULL COMMENT '完成度分数',

`total\_score` decimal(5,2) DEFAULT NULL COMMENT '总分',

`comments` text COMMENT '评语',

`revision\_requirements` text COMMENT '修改要求',

`result` enum('pass','fail','revision\_needed') DEFAULT NULL COMMENT '评定结果',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `unique\_defense\_teacher` (`arrangement\_id`,`teacher\_id`),

KEY `teacher\_id` (`teacher\_id`),

CONSTRAINT `defense\_scores\_ibfk\_1` FOREIGN KEY (`arrangement\_id`) REFERENCES `defense\_arrangements` (`id`),

CONSTRAINT `defense\_scores\_ibfk\_2` FOREIGN KEY (`teacher\_id`) REFERENCES `users` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=19 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='答辩评分及结果表'"

defense\_committee,"CREATE TABLE `defense\_committee` (

`id` int NOT NULL AUTO\_INCREMENT,

`name` varchar(100) NOT NULL COMMENT '答辩委员会名称',

`chairman\_id` varchar(20) NOT NULL COMMENT '主席ID',

`secretary\_id` varchar(20) NOT NULL COMMENT '秘书ID',

`member1\_id` varchar(20) NOT NULL COMMENT '委员1',

`member2\_id` varchar(20) NOT NULL COMMENT '委员2',

`member3\_id` varchar(20) NOT NULL COMMENT '委员3',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

KEY `chairman\_id` (`chairman\_id`),

KEY `secretary\_id` (`secretary\_id`),

KEY `member1\_id` (`member1\_id`),

KEY `member2\_id` (`member2\_id`),

KEY `member3\_id` (`member3\_id`),

CONSTRAINT `defense\_committee\_ibfk\_1` FOREIGN KEY (`chairman\_id`) REFERENCES `users` (`user\_id`),

CONSTRAINT `defense\_committee\_ibfk\_2` FOREIGN KEY (`secretary\_id`) REFERENCES `users` (`user\_id`),

CONSTRAINT `defense\_committee\_ibfk\_3` FOREIGN KEY (`member1\_id`) REFERENCES `users` (`user\_id`),

CONSTRAINT `defense\_committee\_ibfk\_4` FOREIGN KEY (`member2\_id`) REFERENCES `users` (`user\_id`),

CONSTRAINT `defense\_committee\_ibfk\_5` FOREIGN KEY (`member3\_id`) REFERENCES `users` (`user\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='答辩委员会表'"

defense\_arrangements,"CREATE TABLE `defense\_arrangements` (

`id` int NOT NULL AUTO\_INCREMENT,

`student\_id` varchar(50) NOT NULL,

`committee\_id` int NOT NULL,

`defense\_time` datetime NOT NULL COMMENT '答辩时间',

`location` varchar(100) NOT NULL COMMENT '答辩地点',

`duration` int DEFAULT '30' COMMENT '答辩时长(分钟)',

`status` enum('pending','in\_progress','completed','cancelled') DEFAULT 'pending',

`notes` text COMMENT '备注',

`created\_at` datetime DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `unique\_student\_defense` (`student\_id`),

KEY `committee\_id` (`committee\_id`),

CONSTRAINT `defense\_arrangements\_ibfk\_1` FOREIGN KEY (`student\_id`) REFERENCES `students` (`student\_id`),

CONSTRAINT `defense\_arrangements\_ibfk\_2` FOREIGN KEY (`committee\_id`) REFERENCES `defense\_committee` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='答辩安排表'"

before\_defense\_score\_insert,INSERT,defense\_scores,"BEGIN

-- 计算单个评委的总分

SET NEW.total\_score = (NEW.presentation\_score + NEW.qa\_score +

NEW.innovation\_score + NEW.completion\_score) / 4;

-- 设置默认的评定结果

SET NEW.result = IF(NEW.total\_score >= 60, 'pass', 'fail');

END",BEFORE,2024-12-29 21:23:18.90,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

after\_defense\_score\_complete,INSERT,defense\_scores,"BEGIN

DECLARE total\_members INT;

DECLARE scored\_members INT;

-- 获取已评分的委员数量

SELECT COUNT(\*) INTO scored\_members

FROM defense\_scores

WHERE arrangement\_id = NEW.arrangement\_id;

-- 当所有委员都评分完成后，更新答辩状态为已完成

IF scored\_members = 5 THEN

UPDATE defense\_arrangements

SET status = 'completed',

updated\_at = CURRENT\_TIMESTAMP

WHERE id = NEW.arrangement\_id;

END IF;

END",AFTER,2024-12-29 21:23:18.92,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

after\_progress\_record\_insert,INSERT,progress\_records,"BEGIN

IF NEW.score IS NOT NULL THEN

INSERT INTO student\_stage\_scores (

student\_id,

task\_book\_score,

literature\_score,

proposal\_score,

translation\_score,

midterm\_score,

thesis\_submit\_score,

advisor\_review\_score,

peer\_review\_score,

defense\_score

)

VALUES (

NEW.student\_id,

IF(NEW.stage\_type = 'task\_book', NEW.score, NULL),

IF(NEW.stage\_type = 'literature', NEW.score, NULL),

IF(NEW.stage\_type = 'proposal', NEW.score, NULL),

IF(NEW.stage\_type = 'translation', NEW.score, NULL),

IF(NEW.stage\_type = 'midterm', NEW.score, NULL),

IF(NEW.stage\_type = 'thesis\_submit', NEW.score, NULL),

IF(NEW.stage\_type = 'advisor\_review', NEW.score, NULL),

IF(NEW.stage\_type = 'peer\_review', NEW.score, NULL),

IF(NEW.stage\_type = 'defense', NEW.score, NULL)

)

ON DUPLICATE KEY UPDATE

task\_book\_score = IF(NEW.stage\_type = 'task\_book', NEW.score, task\_book\_score),

literature\_score = IF(NEW.stage\_type = 'literature', NEW.score, literature\_score),

proposal\_score = IF(NEW.stage\_type = 'proposal', NEW.score, proposal\_score),

translation\_score = IF(NEW.stage\_type = 'translation', NEW.score, translation\_score),

midterm\_score = IF(NEW.stage\_type = 'midterm', NEW.score, midterm\_score),

thesis\_submit\_score = IF(NEW.stage\_type = 'thesis\_submit', NEW.score, thesis\_submit\_score),

advisor\_review\_score = IF(NEW.stage\_type = 'advisor\_review', NEW.score, advisor\_review\_score),

peer\_review\_score = IF(NEW.stage\_type = 'peer\_review', NEW.score, peer\_review\_score),

defense\_score = IF(NEW.stage\_type = 'defense', NEW.score, defense\_score);

END IF;

END",AFTER,2024-12-29 14:01:24.85,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

after\_task\_book\_record\_insert,INSERT,progress\_records,"BEGIN

-- 只在插入任务书阶段记录时触发

IF NEW.stage\_type = 'task\_book' THEN

-- 从 topic\_selections 表获取相关信息

INSERT INTO task\_assignments (

teacher\_id,

student\_id,

topic\_id,

selection\_id,

title,

content,

requirements,

deadline,

status

)

SELECT

t.teacher\_id, -- 从 topics 表获取教师ID

ts.student\_id, -- 从 topic\_selections 表获取学生ID

ts.topic\_id, -- 从 topic\_selections 表获取课题ID

ts.selection\_id, -- 选题ID

'毕业设计任务书', -- 默认标题

'请填写任务书内容', -- 默认内容

'请按要求完成任务书', -- 默认要求

NEW.deadline, -- 使用进度记录中的截止时间

'draft' -- 初始状态为草稿

FROM topic\_selections ts

JOIN topics t ON ts.topic\_id = t.topic\_id

WHERE ts.student\_id = NEW.student\_id

AND ts.final\_status = 'approved'

LIMIT 1;

END IF;

END",AFTER,2024-12-30 14:43:49.03,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

after\_task\_book\_record\_insert\_scores,INSERT,progress\_records,"BEGIN

-- 只在插入任务书阶段记录时触发

IF NEW.stage\_type = 'task\_book' THEN

-- 检查是否已存在该学生的记录

IF NOT EXISTS (

SELECT 1 FROM student\_stage\_scores

WHERE student\_id = NEW.student\_id

) THEN

-- 插入新的学生阶段分数记录

INSERT INTO student\_stage\_scores (

student\_id,

task\_book\_score,

literature\_score,

proposal\_score,

translation\_score,

midterm\_score,

thesis\_submit\_score,

advisor\_review\_score,

peer\_review\_score,

defense\_score,

total\_weighted\_score

) VALUES (

NEW.student\_id, -- 学生ID

NULL, -- 任务书分数（初始为NULL）

NULL, -- 文献综述分数

NULL, -- 开题报告分数

NULL, -- 外文翻译分数

NULL, -- 中期检查分数

NULL, -- 论文提交分数

NULL, -- 导师评阅分数

NULL, -- 同行评阅分数

NULL, -- 答辩分数

NULL -- 总加权分数

);

END IF;

END IF;

END",AFTER,2024-12-30 14:49:44.44,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

before\_progress\_update,UPDATE,progress\_records,"BEGIN

IF NEW.status = 'completed' AND OLD.status != 'completed' THEN

SET NEW.review\_time = CURRENT\_TIMESTAMP;

END IF;

END",BEFORE,2024-12-29 12:01:53.31,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

after\_peer\_review\_complete,UPDATE,progress\_records,"BEGIN

-- 声明变量用于调试

DECLARE debug\_msg TEXT;

-- 记录触发器被调用的情况

INSERT INTO system\_logs (

action,

details

) VALUES (

'trigger.peer\_review.debug',

CONCAT(

'Trigger called with: ',

'stage\_type=', NEW.stage\_type,

', old\_status=', OLD.status,

', new\_status=', NEW.status,

', student\_id=', NEW.student\_id

)

);

-- 检查是否是同行评议阶段且状态变为已完成

IF NEW.stage\_type = 'peer\_review'

AND NEW.status = 'completed'

AND OLD.status != 'completed' THEN

-- 记录条件匹配的情况

INSERT INTO system\_logs (

action,

details

) VALUES (

'trigger.peer\_review.match',

CONCAT('Conditions matched for student\_id: ', NEW.student\_id)

);

-- 更新对应的论文提交记录状态

UPDATE thesis\_submissions

SET status = 'completed',

updated\_at = CURRENT\_TIMESTAMP

WHERE student\_id = NEW.student\_id

AND status IN ('reviewing', 'review\_assigned', 'submitted') -- 放宽状态检查条件

ORDER BY version DESC

LIMIT 1;

-- 记录更新操作的影响行数

SET debug\_msg = CONCAT(

'Update completed. Rows affected: ', ROW\_COUNT(),

'. Student ID: ', NEW.student\_id

);

INSERT INTO system\_logs (

action,

details

) VALUES (

'trigger.peer\_review.update',

debug\_msg

);

END IF;

END",AFTER,2024-12-29 21:16:53.60,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

after\_progress\_update,UPDATE,progress\_records,"BEGIN

IF NEW.status = 'completed' AND OLD.status != 'completed' AND NEW.stage\_type != 'defense' THEN

CASE NEW.stage\_type

WHEN 'task\_book' THEN

INSERT INTO progress\_stage\_queue (student\_id, next\_stage\_type)

VALUES (NEW.student\_id, 'literature');

WHEN 'literature' THEN

INSERT INTO progress\_stage\_queue (student\_id, next\_stage\_type)

VALUES (NEW.student\_id, 'proposal');

WHEN 'proposal' THEN

INSERT INTO progress\_stage\_queue (student\_id, next\_stage\_type)

VALUES (NEW.student\_id, 'translation');

WHEN 'translation' THEN

INSERT INTO progress\_stage\_queue (student\_id, next\_stage\_type)

VALUES (NEW.student\_id, 'midterm');

WHEN 'midterm' THEN

INSERT INTO progress\_stage\_queue (student\_id, next\_stage\_type)

VALUES (NEW.student\_id, 'thesis\_submit');

WHEN 'thesis\_submit' THEN

INSERT INTO progress\_stage\_queue (student\_id, next\_stage\_type)

VALUES (NEW.student\_id, 'advisor\_review');

WHEN 'advisor\_review' THEN

INSERT INTO progress\_stage\_queue (student\_id, next\_stage\_type)

VALUES (NEW.student\_id, 'peer\_review');

WHEN 'peer\_review' THEN

INSERT INTO progress\_stage\_queue (student\_id, next\_stage\_type)

VALUES (NEW.student\_id, 'defense');

END CASE;

END IF;

END",AFTER,2024-12-30 01:06:31.92,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

sync\_proposal\_progress,UPDATE,progress\_records,"BEGIN

DECLARE v\_teacher\_id VARCHAR(50);

-- 只处理开题报告阶段的更新

IF NEW.stage\_type = 'proposal' THEN

-- 检查是否已存在该学生的开题报告记录

IF EXISTS (SELECT 1 FROM proposal\_reviews WHERE student\_id = NEW.student\_id) THEN

-- 如果存在则更新

UPDATE proposal\_reviews

SET file\_url = CASE

-- 只在文件URL发生变化时更新

WHEN NEW.file\_url != OLD.file\_url THEN NEW.file\_url

ELSE file\_url

END,

status = CASE

-- 只在状态发生变化时更新

WHEN NEW.status != OLD.status THEN NEW.status

ELSE status

END,

submit\_time = CASE

-- 只在提交时间发生变化时更新

WHEN NEW.submit\_time != OLD.submit\_time THEN NEW.submit\_time

ELSE submit\_time

END,

updated\_at = CURRENT\_TIMESTAMP

WHERE student\_id = NEW.student\_id

ORDER BY version DESC

LIMIT 1;

ELSE

-- 如果不存在则获取教师ID并插入新记录

SELECT t.teacher\_id INTO v\_teacher\_id

FROM topic\_selections ts

JOIN topics t ON ts.topic\_id = t.topic\_id

WHERE ts.student\_id = NEW.student\_id

AND ts.final\_status = 'approved'

LIMIT 1;

-- 插入新记录

IF v\_teacher\_id IS NOT NULL THEN

INSERT INTO proposal\_reviews (

student\_id,

teacher\_id,

version,

file\_url,

status,

submit\_time,

created\_at,

updated\_at

) VALUES (

NEW.student\_id,

v\_teacher\_id,

1,

NEW.file\_url,

NEW.status,

NEW.submit\_time,

CURRENT\_TIMESTAMP,

CURRENT\_TIMESTAMP

);

END IF;

END IF;

END IF;

END",AFTER,2024-12-30 16:38:51.12,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

sync\_translation\_progress,UPDATE,progress\_records,"BEGIN

-- 只处理外文翻译阶段的更新

IF NEW.stage\_type = 'translation' THEN

-- 检查是否已存在该学生的外文翻译记录

IF EXISTS (SELECT 1 FROM translation\_reviews WHERE student\_id = NEW.student\_id) THEN

-- 如果存在则更新

UPDATE translation\_reviews

SET file\_url = CASE

-- 只在文件URL发生变化时更新

WHEN NEW.file\_url != OLD.file\_url THEN NEW.file\_url

ELSE file\_url

END,

status = CASE

-- 只在状态发生变化时更新

WHEN NEW.status != OLD.status THEN

CASE NEW.status

WHEN 'not\_started' THEN 'pending'

WHEN 'in\_progress' THEN 'pending'

WHEN 'reviewing' THEN 'reviewing'

WHEN 'revision\_needed' THEN 'revision\_needed'

WHEN 'completed' THEN 'completed'

ELSE status

END

ELSE status

END,

submit\_time = CASE

-- 只在提交时间发生变化时更新

WHEN NEW.submit\_time != OLD.submit\_time THEN NEW.submit\_time

ELSE submit\_time

END,

updated\_at = CURRENT\_TIMESTAMP

WHERE student\_id = NEW.student\_id

ORDER BY version DESC

LIMIT 1;

ELSE

-- 如果不存在则插入新记录

INSERT INTO translation\_reviews (

student\_id,

version,

file\_url,

status,

submit\_time,

created\_at,

updated\_at

) VALUES (

NEW.student\_id,

1,

NEW.file\_url,

CASE NEW.status

WHEN 'not\_started' THEN 'pending'

WHEN 'in\_progress' THEN 'pending'

WHEN 'reviewing' THEN 'reviewing'

WHEN 'revision\_needed' THEN 'revision\_needed'

WHEN 'completed' THEN 'completed'

ELSE 'pending'

END,

NEW.submit\_time,

CURRENT\_TIMESTAMP,

CURRENT\_TIMESTAMP

);

END IF;

END IF;

END",AFTER,2024-12-30 16:54:00.00,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

update\_stage\_score,UPDATE,progress\_records,"BEGIN

-- 只要成绩发生变化就更新

IF NEW.score != OLD.score OR (NEW.score IS NOT NULL AND OLD.score IS NULL) THEN

-- 先检查是否存在记录

IF EXISTS (SELECT 1 FROM student\_stage\_scores WHERE student\_id = NEW.student\_id) THEN

-- 如果存在记录则更新

WITH stage\_weights AS (

SELECT type, weight

FROM progress\_stages

WHERE status = 'active'

)

UPDATE student\_stage\_scores ss

SET

-- 更新对应阶段分数

task\_book\_score = IF(NEW.stage\_type = 'task\_book', NEW.score, ss.task\_book\_score),

literature\_score = IF(NEW.stage\_type = 'literature', NEW.score, ss.literature\_score),

proposal\_score = IF(NEW.stage\_type = 'proposal', NEW.score, ss.proposal\_score),

translation\_score = IF(NEW.stage\_type = 'translation', NEW.score, ss.translation\_score),

midterm\_score = IF(NEW.stage\_type = 'midterm', NEW.score, ss.midterm\_score),

thesis\_submit\_score = IF(NEW.stage\_type = 'thesis\_submit', NEW.score, ss.thesis\_submit\_score),

advisor\_review\_score = IF(NEW.stage\_type = 'advisor\_review', NEW.score, ss.advisor\_review\_score),

peer\_review\_score = IF(NEW.stage\_type = 'peer\_review', NEW.score, ss.peer\_review\_score),

defense\_score = IF(NEW.stage\_type = 'defense', NEW.score, ss.defense\_score),

-- 计算总加权分数

total\_weighted\_score = (

COALESCE(IF(NEW.stage\_type = 'task\_book', NEW.score, ss.task\_book\_score) \*

(SELECT weight FROM stage\_weights WHERE type = 'task\_book'), 0) +

COALESCE(IF(NEW.stage\_type = 'literature', NEW.score, ss.literature\_score) \*

(SELECT weight FROM stage\_weights WHERE type = 'literature'), 0) +

COALESCE(IF(NEW.stage\_type = 'proposal', NEW.score, ss.proposal\_score) \*

(SELECT weight FROM stage\_weights WHERE type = 'proposal'), 0) +

COALESCE(IF(NEW.stage\_type = 'translation', NEW.score, ss.translation\_score) \*

(SELECT weight FROM stage\_weights WHERE type = 'translation'), 0) +

COALESCE(IF(NEW.stage\_type = 'midterm', NEW.score, ss.midterm\_score) \*

(SELECT weight FROM stage\_weights WHERE type = 'midterm'), 0) +

COALESCE(IF(NEW.stage\_type = 'thesis\_submit', NEW.score, ss.thesis\_submit\_score) \*

(SELECT weight FROM stage\_weights WHERE type = 'thesis\_submit'), 0) +

COALESCE(IF(NEW.stage\_type = 'advisor\_review', NEW.score, ss.advisor\_review\_score) \*

(SELECT weight FROM stage\_weights WHERE type = 'advisor\_review'), 0) +

COALESCE(IF(NEW.stage\_type = 'peer\_review', NEW.score, ss.peer\_review\_score) \*

(SELECT weight FROM stage\_weights WHERE type = 'peer\_review'), 0) +

COALESCE(IF(NEW.stage\_type = 'defense', NEW.score, ss.defense\_score) \*

(SELECT weight FROM stage\_weights WHERE type = 'defense'), 0)

) / 100

WHERE ss.student\_id = NEW.student\_id;

ELSE

-- 如果记录不存在则插入

INSERT INTO student\_stage\_scores (

student\_id,

task\_book\_score,

literature\_score,

proposal\_score,

translation\_score,

midterm\_score,

thesis\_submit\_score,

advisor\_review\_score,

peer\_review\_score,

defense\_score,

total\_weighted\_score

)

SELECT

NEW.student\_id,

IF(NEW.stage\_type = 'task\_book', NEW.score, NULL),

IF(NEW.stage\_type = 'literature', NEW.score, NULL),

IF(NEW.stage\_type = 'proposal', NEW.score, NULL),

IF(NEW.stage\_type = 'translation', NEW.score, NULL),

IF(NEW.stage\_type = 'midterm', NEW.score, NULL),

IF(NEW.stage\_type = 'thesis\_submit', NEW.score, NULL),

IF(NEW.stage\_type = 'advisor\_review', NEW.score, NULL),

IF(NEW.stage\_type = 'peer\_review', NEW.score, NULL),

IF(NEW.stage\_type = 'defense', NEW.score, NULL),

(SELECT weight FROM progress\_stages WHERE type = NEW.stage\_type AND status = 'active') \* NEW.score / 100;

END IF;

END IF;

END",AFTER,2024-12-30 17:05:34.42,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

sync\_midterm\_progress,UPDATE,progress\_records,"BEGIN

-- 只处理中期检查阶段的更新

IF NEW.stage\_type = 'midterm' THEN

-- 检查是否已存在该学生的中期检查记录

IF EXISTS (SELECT 1 FROM midterm\_reports WHERE student\_id = NEW.student\_id) THEN

-- 如果存在则更新

UPDATE midterm\_reports

SET file\_url = CASE

-- 只在文件URL发生变化时更新

WHEN NEW.file\_url != OLD.file\_url THEN NEW.file\_url

ELSE file\_url

END,

status = CASE

-- 只在状态发生变化时更新

WHEN NEW.status != OLD.status THEN

CASE NEW.status

WHEN 'not\_started' THEN 'pending'

WHEN 'in\_progress' THEN 'pending'

WHEN 'reviewing' THEN 'reviewing'

WHEN 'revision\_needed' THEN 'revision\_needed'

WHEN 'completed' THEN 'completed'

ELSE status

END

ELSE status

END,

submit\_time = CASE

-- 只在提交时间发生变化时更新

WHEN NEW.submit\_time != OLD.submit\_time THEN NEW.submit\_time

ELSE submit\_time

END,

updated\_at = CURRENT\_TIMESTAMP

WHERE student\_id = NEW.student\_id

ORDER BY version DESC

LIMIT 1;

ELSE

-- 如果不存在则插入新记录

INSERT INTO midterm\_reports (

student\_id,

version,

file\_url,

status,

submit\_time,

created\_at,

updated\_at

) VALUES (

NEW.student\_id,

1,

NEW.file\_url,

CASE NEW.status

WHEN 'not\_started' THEN 'pending'

WHEN 'in\_progress' THEN 'pending'

WHEN 'reviewing' THEN 'reviewing'

WHEN 'revision\_needed' THEN 'revision\_needed'

WHEN 'completed' THEN 'completed'

ELSE 'pending'

END,

NEW.submit\_time,

CURRENT\_TIMESTAMP,

CURRENT\_TIMESTAMP

);

END IF;

END IF;

END",AFTER,2024-12-30 17:08:03.84,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

sync\_thesis\_progress,UPDATE,progress\_records,"BEGIN

-- 只处理论文提交阶段的更新

IF NEW.stage\_type = 'thesis\_submit' THEN

-- 检查是否已存在该学生的论文提交记录

IF EXISTS (SELECT 1 FROM thesis\_submissions WHERE student\_id = NEW.student\_id) THEN

UPDATE thesis\_submissions

SET

file\_url = CASE

WHEN NEW.file\_url != OLD.file\_url THEN NEW.file\_url

ELSE file\_url

END,

status = CASE NEW.status

WHEN 'not\_started' THEN 'submitted'

WHEN 'in\_progress' THEN 'submitted'

WHEN 'reviewing' THEN 'advisor\_reviewing'

WHEN 'revision\_needed' THEN 'revision\_needed'

WHEN 'completed' THEN 'advisor\_approved'

ELSE status

END,

advisor\_review\_status = CASE NEW.status

WHEN 'completed' THEN 'approved'

WHEN 'reviewing' THEN 'reviewing'

WHEN 'revision\_needed' THEN 'rejected'

ELSE advisor\_review\_status

END,

advisor\_review\_time = CASE

WHEN NEW.status IN ('completed', 'reviewing', 'revision\_needed')

AND NEW.status != OLD.status

THEN CURRENT\_TIMESTAMP

ELSE advisor\_review\_time

END,

submit\_time = CASE

WHEN NEW.submit\_time != OLD.submit\_time THEN NEW.submit\_time

ELSE submit\_time

END,

updated\_at = CURRENT\_TIMESTAMP

WHERE student\_id = NEW.student\_id;

ELSE

-- 如果不存在则插入新记录

INSERT INTO thesis\_submissions (

student\_id,

version,

file\_url,

title,

status,

advisor\_review\_status,

submit\_time,

created\_at,

updated\_at

) VALUES (

NEW.student\_id,

1,

NEW.file\_url,

'论文题目待更新',

CASE NEW.status

WHEN 'not\_started' THEN 'submitted'

WHEN 'in\_progress' THEN 'submitted'

WHEN 'reviewing' THEN 'advisor\_reviewing'

WHEN 'revision\_needed' THEN 'revision\_needed'

WHEN 'completed' THEN 'advisor\_approved'

ELSE 'submitted'

END,

CASE NEW.status

WHEN 'completed' THEN 'approved'

WHEN 'reviewing' THEN 'reviewing'

WHEN 'revision\_needed' THEN 'rejected'

ELSE 'pending'

END,

NEW.submit\_time,

CURRENT\_TIMESTAMP,

CURRENT\_TIMESTAMP

);

END IF;

END IF;

END",AFTER,2024-12-30 19:16:01.06,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

proposal\_review\_version\_update,UPDATE,proposal\_reviews,"BEGIN

-- 当状态变更为 revision\_needed 时，版本号加1

IF NEW.status = 'revision\_needed' AND OLD.status != 'revision\_needed' THEN

SET NEW.version = OLD.version + 1;

END IF;

END",BEFORE,2024-12-18 20:38:34.01,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

update\_progress\_on\_task\_update,UPDATE,task\_assignments,"BEGIN

-- 当任务书被更新时（不需要检查具体字段）

IF NEW.updated\_at != OLD.updated\_at THEN

-- 只更新对应的进度记录状态为 in\_progress

UPDATE progress\_records

SET status = 'in\_progress',

deadline = NEW.deadline,

updated\_at = NOW()

WHERE student\_id = NEW.student\_id

AND stage\_type = 'task\_book';

END IF;

END",AFTER,2024-12-30 15:46:07.48,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

after\_topic\_selection\_approved,UPDATE,topic\_selections,"BEGIN

-- 检查 final\_status 是否变为 approved

IF NEW.final\_status = 'approved' AND OLD.final\_status != 'approved' THEN

-- 插入任务书阶段记录

INSERT INTO progress\_records (

student\_id,

stage\_type,

status,

deadline

) VALUES (

NEW.student\_id,

'task\_book',

'not\_started',

DATE\_ADD(CURRENT\_TIMESTAMP, INTERVAL 7 DAY) -- 设置截止时间为当前时间后7天

);

END IF;

END",AFTER,2024-12-30 14:34:33.08,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci

translation\_review\_version\_update,UPDATE,translation\_reviews,"BEGIN

-- 当状态变更为 revision\_needed 时，版本号加1

IF NEW.status = 'revision\_needed' AND OLD.status != 'revision\_needed' THEN

SET NEW.version = OLD.version + 1;

END IF;

END",BEFORE,2024-12-18 22:19:15.98,"ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION",root@localhost,utf8mb4,utf8mb4\_0900\_ai\_ci,utf8mb4\_0900\_ai\_ci  
  
defense\_arrangements,committee\_id,committee\_id,1,BTREE

defense\_arrangements,PRIMARY,id,0,BTREE

defense\_arrangements,unique\_student\_defense,student\_id,0,BTREE

defense\_committee,chairman\_id,chairman\_id,1,BTREE

defense\_committee,member1\_id,member1\_id,1,BTREE

defense\_committee,member2\_id,member2\_id,1,BTREE

defense\_committee,member3\_id,member3\_id,1,BTREE

defense\_committee,PRIMARY,id,0,BTREE

defense\_committee,secretary\_id,secretary\_id,1,BTREE

defense\_scores,idx\_arrangement\_total,arrangement\_id,1,BTREE

defense\_scores,idx\_arrangement\_total,total\_score,1,BTREE

defense\_scores,idx\_score\_stats,total\_score,1,BTREE

defense\_scores,idx\_score\_stats,created\_at,1,BTREE

defense\_scores,PRIMARY,id,0,BTREE

defense\_scores,teacher\_id,teacher\_id,1,BTREE

defense\_scores,unique\_defense\_teacher,arrangement\_id,0,BTREE

defense\_scores,unique\_defense\_teacher,teacher\_id,0,BTREE

literature\_reviews,idx\_student\_status,student\_id,1,BTREE

literature\_reviews,idx\_student\_status,status,1,BTREE

literature\_reviews,idx\_teacher,teacher\_id,1,BTREE

literature\_reviews,PRIMARY,id,0,BTREE

literature\_reviews,unique\_student\_version,student\_id,0,BTREE

literature\_reviews,unique\_student\_version,version,0,BTREE

midterm\_reports,idx\_student\_status,student\_id,1,BTREE

midterm\_reports,idx\_student\_status,status,1,BTREE

midterm\_reports,idx\_teacher\_id,teacher\_id,1,BTREE

midterm\_reports,PRIMARY,id,0,BTREE

next\_stage\_queue,PRIMARY,id,0,BTREE

next\_stage\_queue,processed,processed,1,BTREE

progress\_records,idx\_status\_deadline,status,1,BTREE

progress\_records,idx\_status\_deadline,deadline,1,BTREE

progress\_records,PRIMARY,id,0,BTREE

progress\_records,unique\_student\_stage,student\_id,0,BTREE

progress\_records,unique\_student\_stage,stage\_type,0,BTREE

progress\_stage\_queue,PRIMARY,id,0,BTREE

progress\_stages,PRIMARY,id,0,BTREE

progress\_stages,sequence,sequence,0,BTREE

proposal\_reviews,PRIMARY,id,0,BTREE

proposal\_reviews,student\_id,student\_id,1,BTREE

proposal\_reviews,teacher\_id,teacher\_id,1,BTREE

student\_stage\_scores,PRIMARY,id,0,BTREE

student\_stage\_scores,student\_id,student\_id,1,BTREE

students,PRIMARY,student\_id,0,BTREE

supervisors,idx\_department,department,1,BTREE

supervisors,PRIMARY,supervisor\_id,0,BTREE

system\_logs,idx\_created\_at,created\_at,1,BTREE

system\_logs,idx\_user\_action\_time,user\_id,1,BTREE

system\_logs,idx\_user\_action\_time,action,1,BTREE

system\_logs,idx\_user\_action\_time,created\_at,1,BTREE

system\_logs,PRIMARY,log\_id,0,BTREE

system\_messages,created\_by,created\_by,1,BTREE

system\_messages,idx\_message\_status\_priority,status,1,BTREE

system\_messages,idx\_message\_status\_priority,priority,1,BTREE

system\_messages,PRIMARY,message\_id,0,BTREE

task\_assignments,PRIMARY,task\_id,0,BTREE

task\_assignments,selection\_id,selection\_id,1,BTREE

task\_assignments,student\_id,student\_id,1,BTREE

task\_assignments,teacher\_id,teacher\_id,1,BTREE

task\_assignments,topic\_id,topic\_id,1,BTREE

thesis\_review\_details,assignment\_id,assignment\_id,1,BTREE

thesis\_review\_details,PRIMARY,id,0,BTREE

thesis\_reviews\_assignment,idx\_reviewer\_status,reviewer\_id,1,BTREE

thesis\_reviews\_assignment,idx\_reviewer\_status,status,1,BTREE

thesis\_reviews\_assignment,PRIMARY,id,0,BTREE

thesis\_reviews\_assignment,reviewer\_id,reviewer\_id,1,BTREE

thesis\_reviews\_assignment,student\_id,student\_id,1,BTREE

thesis\_submissions,advisor\_id,advisor\_id,1,BTREE

thesis\_submissions,PRIMARY,id,0,BTREE

thesis\_submissions,student\_id,student\_id,1,BTREE

topic\_selections,idx\_student\_topic\_status,student\_id,1,BTREE

topic\_selections,idx\_student\_topic\_status,topic\_id,1,BTREE

topic\_selections,idx\_student\_topic\_status,final\_status,1,BTREE

topic\_selections,idx\_topic\_status,topic\_id,1,BTREE

topic\_selections,idx\_topic\_status,final\_status,1,BTREE

topic\_selections,PRIMARY,selection\_id,0,BTREE

topic\_selections,topic\_selections\_ibfk\_3,director\_id,1,BTREE

topics,idx\_teacher\_status,teacher\_id,1,BTREE

topics,idx\_teacher\_status,status,1,BTREE

topics,PRIMARY,topic\_id,0,BTREE

translation\_reviews,idx\_student\_status,student\_id,1,BTREE

translation\_reviews,idx\_student\_status,status,1,BTREE

translation\_reviews,PRIMARY,id,0,BTREE

users,idx\_role,role,1,BTREE

users,idx\_staff\_id,staff\_id,0,BTREE

users,idx\_student\_id,student\_id,0,BTREE

users,idx\_username\_status,username,1,BTREE

users,idx\_username\_status,status,1,BTREE

users,PRIMARY,user\_id,0,BTREE

v\_defense\_arrangement\_details,"select `da`.`id` AS `arrangement\_id`,`u1`.`name` AS `student\_name`,`da`.`defense\_time` AS `defense\_time`,`da`.`location` AS `location`,`u2`.`name` AS `chairman\_name`,`u3`.`name` AS `secretary\_name`,`da`.`status` AS `status` from ((((`graduation\_management`.`defense\_arrangements` `da` join `graduation\_management`.`users` `u1` on((`da`.`student\_id` = `u1`.`user\_id`))) join `graduation\_management`.`defense\_committee` `dc` on((`da`.`committee\_id` = `dc`.`id`))) join `graduation\_management`.`users` `u2` on((`dc`.`chairman\_id` = `u2`.`user\_id`))) join `graduation\_management`.`users` `u3` on((`dc`.`secretary\_id` = `u3`.`user\_id`)))",YES,root@localhost,DEFINER,NONE

v\_defense\_score\_summary,"select `ds`.`arrangement\_id` AS `arrangement\_id`,`u1`.`name` AS `student\_name`,avg(`ds`.`total\_score`) AS `average\_score`,min(`ds`.`total\_score`) AS `min\_score`,max(`ds`.`total\_score`) AS `max\_score`,count(`ds`.`id`) AS `scorer\_count` from ((`graduation\_management`.`defense\_scores` `ds` join `graduation\_management`.`defense\_arrangements` `da` on((`ds`.`arrangement\_id` = `da`.`id`))) join `graduation\_management`.`users` `u1` on((`da`.`student\_id` = `u1`.`user\_id`))) group by `ds`.`arrangement\_id`",NO,root@localhost,DEFINER,NONE

v\_stage\_completion\_stats,"select `ps`.`type` AS `type`,`ps`.`name` AS `stage\_name`,count(`pr`.`id`) AS `total\_students`,sum((case when (`pr`.`status` = 'completed') then 1 else 0 end)) AS `completed\_count`,sum((case when (`pr`.`status` = 'reviewing') then 1 else 0 end)) AS `reviewing\_count`,sum((case when (`pr`.`status` = 'rejected') then 1 else 0 end)) AS `rejected\_count` from (`graduation\_management`.`progress\_stages` `ps` left join `graduation\_management`.`progress\_records` `pr` on((`ps`.`type` = `pr`.`stage\_type`))) group by `ps`.`type`",NO,root@localhost,DEFINER,NONE

v\_student\_comprehensive\_scores,"select `pr`.`student\_id` AS `student\_id`,`u`.`name` AS `student\_name`,max((case when (`pr`.`stage\_type` = 'task\_book') then `pr`.`score` end)) AS `task\_book\_score`,max((case when (`pr`.`stage\_type` = 'literature') then `pr`.`score` end)) AS `literature\_score`,max((case when (`pr`.`stage\_type` = 'proposal') then `pr`.`score` end)) AS `proposal\_score`,max((case when (`pr`.`stage\_type` = 'translation') then `pr`.`score` end)) AS `translation\_score`,max((case when (`pr`.`stage\_type` = 'midterm') then `pr`.`score` end)) AS `midterm\_score`,max((case when (`pr`.`stage\_type` = 'thesis\_submit') then `pr`.`score` end)) AS `thesis\_score`,avg(`ds`.`total\_score`) AS `defense\_score`,round((((coalesce(max((case when (`pr`.`stage\_type` = 'task\_book') then `pr`.`score` end)),0) \* `ps1`.`weight`) + (coalesce(max((case when (`pr`.`stage\_type` = 'defense') then `ds`.`total\_score` end)),0) \* `ps2`.`weight`)) / (`ps1`.`weight` + `ps2`.`weight`)),2) AS `weighted\_total\_score` from (((((`graduation\_management`.`progress\_records` `pr` join `graduation\_management`.`users` `u` on((`pr`.`student\_id` = `u`.`user\_id`))) left join `graduation\_management`.`progress\_stages` `ps1` on((`pr`.`stage\_type` = `ps1`.`type`))) left join `graduation\_management`.`defense\_arrangements` `da` on((`pr`.`student\_id` = `da`.`student\_id`))) left join `graduation\_management`.`defense\_scores` `ds` on((`da`.`id` = `ds`.`arrangement\_id`))) left join `graduation\_management`.`progress\_stages` `ps2` on((`ps2`.`type` = 'defense'))) group by `pr`.`student\_id`",NO,root@localhost,DEFINER,NONE

v\_student\_performance\_summary,"select `s`.`student\_id` AS `student\_id`,`u`.`name` AS `student\_name`,count(`pr`.`id`) AS `completed\_stages`,avg(`pr`.`score`) AS `avg\_progress\_score`,`ts`.`title` AS `thesis\_title`,`ds`.`total\_score` AS `defense\_score` from (((((`graduation\_management`.`students` `s` join `graduation\_management`.`users` `u` on((`s`.`student\_id` = `u`.`user\_id`))) left join `graduation\_management`.`progress\_records` `pr` on((`s`.`student\_id` = `pr`.`student\_id`))) left join `graduation\_management`.`thesis\_submissions` `ts` on((`s`.`student\_id` = `ts`.`student\_id`))) left join `graduation\_management`.`defense\_arrangements` `da` on((`s`.`student\_id` = `da`.`student\_id`))) left join `graduation\_management`.`defense\_scores` `ds` on((`da`.`id` = `ds`.`arrangement\_id`))) group by `s`.`student\_id`",NO,root@localhost,DEFINER,NONE

v\_student\_progress\_overview,"select `pr`.`student\_id` AS `student\_id`,`u`.`name` AS `student\_name`,`ps`.`name` AS `stage\_name`,`pr`.`status` AS `status`,`pr`.`score` AS `score`,`pr`.`submit\_time` AS `submit\_time`,`pr`.`deadline` AS `deadline`,(case when (`pr`.`submit\_time` > `pr`.`deadline`) then 'delayed' when ((`pr`.`submit\_time` is null) and (now() > `pr`.`deadline`)) then 'overdue' else 'normal' end) AS `time\_status` from ((`graduation\_management`.`progress\_records` `pr` join `graduation\_management`.`users` `u` on((`pr`.`student\_id` = `u`.`user\_id`))) join `graduation\_management`.`progress\_stages` `ps` on((`pr`.`stage\_type` = `ps`.`type`)))",YES,root@localhost,DEFINER,NONE

v\_teacher\_supervision\_stats,"select `t`.`topic\_id` AS `topic\_id`,`u`.`name` AS `teacher\_name`,count(distinct `ts`.`student\_id`) AS `student\_count`,avg(`ds`.`total\_score`) AS `avg\_defense\_score` from ((((`graduation\_management`.`topics` `t` join `graduation\_management`.`users` `u` on((`t`.`teacher\_id` = `u`.`user\_id`))) left join `graduation\_management`.`topic\_selections` `ts` on((`t`.`topic\_id` = `ts`.`topic\_id`))) left join `graduation\_management`.`defense\_arrangements` `da` on((`ts`.`student\_id` = `da`.`student\_id`))) left join `graduation\_management`.`defense\_scores` `ds` on((`da`.`id` = `ds`.`arrangement\_id`))) group by `t`.`topic\_id`",NO,root@localhost,DEFINER,NONE

v\_thesis\_review\_assignments,"select `tra`.`id` AS `assignment\_id`,`tra`.`student\_id` AS `student\_id`,`u1`.`name` AS `student\_name`,`u2`.`name` AS `reviewer\_name`,`ts`.`title` AS `thesis\_title`,`tra`.`review\_type` AS `review\_type`,`tra`.`status` AS `status`,`tra`.`assigned\_at` AS `assigned\_at`,`tra`.`completed\_at` AS `completed\_at` from (((`graduation\_management`.`thesis\_reviews\_assignment` `tra` join `graduation\_management`.`users` `u1` on((`tra`.`student\_id` = `u1`.`user\_id`))) join `graduation\_management`.`users` `u2` on((`tra`.`reviewer\_id` = `u2`.`user\_id`))) join `graduation\_management`.`thesis\_submissions` `ts` on((`tra`.`student\_id` = `ts`.`student\_id`)))",YES,root@localhost,DEFINER,NONE

v\_thesis\_review\_summary,"select `trd`.`assignment\_id` AS `assignment\_id`,`tra`.`student\_id` AS `student\_id`,`u`.`name` AS `student\_name`,avg(`trd`.`total\_score`) AS `average\_score`,group\_concat(distinct `trd`.`review\_comments` separator ',') AS `all\_comments` from ((`graduation\_management`.`thesis\_review\_details` `trd` join `graduation\_management`.`thesis\_reviews\_assignment` `tra` on((`trd`.`assignment\_id` = `tra`.`id`))) join `graduation\_management`.`users` `u` on((`tra`.`student\_id` = `u`.`user\_id`))) group by `trd`.`assignment\_id`",NO,root@localhost,DEFINER,NONE

v\_topic\_selection\_status,"select `t`.`topic\_id` AS `topic\_id`,`t`.`title` AS `title`,`t`.`type` AS `type`,`u`.`name` AS `teacher\_name`,`t`.`status` AS `topic\_status`,count(`ts`.`selection\_id`) AS `selected\_count`,sum((case when (`ts`.`final\_status` = 'approved') then 1 else 0 end)) AS `approved\_count` from ((`graduation\_management`.`topics` `t` left join `graduation\_management`.`users` `u` on((`t`.`teacher\_id` = `u`.`user\_id`))) left join `graduation\_management`.`topic\_selections` `ts` on((`t`.`topic\_id` = `ts`.`topic\_id`))) group by `t`.`topic\_id`",NO,root@localhost,DEFINER,NONE