# پروپوزال پروژه: سیستم مدیریت انبار با پایتون

## ۱. مقدمه

این سند، برای یک **سیستم مدیریت انبار (Inventory Management System)** است که با استفاده از زبان برنامه‌نویسی پایتون توسعه یافته است. هدف اصلی این پروژه، ارائه یک راه‌حل ساده، کارآمد و کاربردی برای مدیریت موجودی محصولات در کسب‌وکارهای کوچک، مانند فروشگاه‌های مواد غذایی یا خرده‌فروشی‌ها است. این سیستم به صورت دستی عمل می‌کند اما با ساختاری سازمان‌یافته، فرآیندهای ثبت، ویرایش، حذف و گزارش‌گیری از محصولات را بهینه‌سازی می‌کند. برای ذخیره‌سازی داده‌ها از فایلCSV استفاده شده و قابلیت خروجی گرفتن با فرمت Excel نیز برای تحلیل و دسترسی آسان‌تر فراهم شده است.

## ۲. معماری و ساختار پروژه

ساختار فایل‌های پروژه به شکل زیر طراحی شده است تا منطق برنامه و داده‌ها از یکدیگر تفکیک شوند:

IMS\_project/  
│  
├── inventory.py # فایل اصلی اجرای برنامه  
├── products.csv # پایگاه داده اولیه محصولات (فایل اصلی)  
├── session\_data.csv # فایل موقت برای ذخیره تغییرات در طول اجرای برنامه  
└── inventory\_export.xlsx # خروجی اکسل که توسط برنامه تولید می‌شود

### ۳. شرح قابلیت‌های اصلی سیستم

سیستم از مجموعه‌ای از کلاس‌ها و توابع تشکیل شده است که هر یک وظیفه مشخصی را بر عهده دارند.

**۳.۱. مدیریت داده محصولات (کلاس Product)**

برای نگهداری اطلاعات هر محصول، یک کلاس به نام Product تعریف شده است که ویژگی‌های زیر را مدیریت می‌کند:

* **:code** کد منحصر به فرد محصول
* **name** : نام محصول
* **:price** قیمت
* **:quantity** تعداد موجودی

این کلاس همچنین دارای متدی برای نمایش اطلاعات کامل یک محصول است.

**۳.۲. بارگذاری و ذخیره‌سازی داده‌ها**

* **load\_from\_csv()**: در ابتدای اجرای برنامه، این تابع اطلاعات را از فایل اصلی products.csv خوانده و آن‌ها را به لیستی از اشیاء Product تبدیل می‌کند.
* **save\_to\_csv()**: برای جلوگیری از دستکاری ناخواسته فایل اصلی، تمام تغییرات (افزودن، ویرایش، حذف) در حین اجرای برنامه، در فایل موقت session\_data.csv ذخیره می‌شوند.

**۳.۳. عملیات اصلی روی محصولات (CRUD)**

* **افزودن محصول :(add\_product)** اطلاعات محصول جدید (نام، قیمت و تعداد) از کاربر دریافت شده، یک شیء جدید از آن ساخته و به لیست موجودی اضافه می‌شود.
* **ویرایش محصول :(edit\_product)** کاربر با وارد کردن کد یا نام محصول، آن را پیدا کرده و سپس می‌تواند هر یک از مشخصات آن را ویرایش کند.
* **حذف محصول :(delete\_product)** پس از جستجو و یافتن محصول، اطلاعات آن به کاربر نمایش داده می‌شود و پس از دریافت تأیید نهایی، محصول از لیست حذف می‌گردد.
* **نمایش همه محصولات :(show\_all\_products)** تمام محصولات موجود در انبار به همراه جزئیات کامل آن‌ها نمایش داده می‌شود.

**۳.۴. ابزارهای کاربردی و گزارش‌گیری**

* **جستجوی محصول :(search\_product)** امکان جستجوی سریع یک محصول بر اساس نام یا کد آن را فراهم می‌کند.
* **مرتب‌سازی محصولات :(sort\_products)** به کاربر اجازه می‌دهد لیست محصولات را بر اساس **نام**، **قیمت** یا **تعداد موجودی** مرتب کرده و نتیجه را مشاهده کند.
* **خروجی اکسل :(export\_to\_excel)** با استفاده از کتابخانه pandas، یک خروجی با فرمت .xlsxاز آخرین وضعیت موجودی انبار تهیه کرده و به صورت خودکار آن را در نرم‌افزار Excel باز می‌کند.

۴. رابط کاربری و جریان برنامه

برنامه از طریق یک منوی تعاملی در ترمینال کنترل می‌شود. این منو در یک حلقه بی‌نهایت (while True) اجرا شده و گزینه‌های زیر را در اختیار کاربر قرار می‌دهد:

1. **نمایش همه محصولات**
2. **افزودن محصول**
3. **ویرایش محصول**
4. **حذف محصول**
5. **مرتب‌سازی محصولات**
6. **جستجوی محصول**
7. **گرفتن خروجی Excel**
8. **ذخیره نهایی تغییرات**
9. **خروج از برنامه**

**مدیریت ذخیره‌سازی و خروج امن**

* **:confirm\_save\_changes()**در پایان کار، هنگام انتخاب گزینه "خروج" یا "ذخیره تغییرات"، برنامه از کاربر می‌پرسد که آیا مایل است تغییرات اعمال‌شده در session\_data.csv را به فایل اصلی (products.csv) منتقل کند یا خیر. این ویژگی از اعمال تغییرات ناخواسته جلوگیری می‌کند.
* **مدیریت خروج اضطراری :(KeyboardInterrupt)** اگر کاربر با فشردن کلیدهای Ctrl+Cقصد خروج ناگهانی از برنامه را داشته باشد، سیستم این رویداد را مدیریت کرده و پیش از بسته شدن، مجدداً فرصت ذخیره تغییرات را به او می‌دهد.

۵. نتیجه‌گیری

پروژه **سیستم مدیریت انبار** یک ابزار ساده، مستقل و قابل گسترش است که برای کسب‌وکارهای کوچک طراحی شده است. عدم وابستگی به پایگاه‌داده‌های پیچیده و استفاده از فایل‌های متنی رایج مانند CSV و Excel، کار با آن را برای کاربران غیرفنی نیز آسان می‌سازد.