

# ভাষার ব্যবধান : ডিবেজ-৩+ হতে ডিবেজ-৪

একটি ডিক সিলকা (রবিন)

## শেষ পর্ব

পূর্বে প্রকাশিত পর্বসমূহে ডিবেজ-৪ এর সর্বাপেক্ষা তরুণতম সীকারগুলো নিয়ে আলোচনার পর এবার আমরা মনোযোগ সেরা কিছুভাবে ডিবেজ-৩ হতে পুরোনো প্রোগ্রাম কোডগুলো নতুন পরিবেশে স্থানান্তর করা যাক। ডিবেজ-৩+ এর সমস্ত এপ্লিকেশন ডিবেজ-৪ এ শব্দেত্র একত্রিত করা পদ্ধতিসহ। পুরোনো প্রোগ্রামকে ডিবেজ-৪ এর অধীনে রান করলেই হল। এটি তখন ডিবেজ-৪ এর অবলম্বিত কোড হয়ে যাবে।

এ পথ যদিও খুবই সহজ, তবে এরূপ পদ্ধতি নিয়ে আপনি কিছু ডিবেজ-৪ এর নতুন নতুন সীকারগুলোর সুবিধা গ্রহণ হতে বাধ্য হবেন; তাই অধিক পছন্দনীয় স্ট্রাটেজি হলে এপ্লিকেশন কোডকে ডিবেজ-৪ এ এমনভাবে স্থানান্তর করা যাতে এপ্লিকেশনকে এমনভাবে কিছু পরিবর্তন করতে হবে যেন তা ডিবেজ-৪ এর নিষেধ করার দ্বারা উপস্থিত হয়। এই পরিবর্তনের (modifications) ফলে প্রোগ্রাম এপ্লিকেশন পতি বৃদ্ধি পাবে। কোড ভলিউমের আকার হ্রাস পাবে। ব্যবহারকারী যা ক্রোধের নিকট প্রোগ্রামগুলো আরো ভালভাবে বিতরণ করা সম্ভব হবে এবং সর্বোপরি ভবিষ্যতে প্রোগ্রাম রক্ষণাবেক্ষণ আরও সহজ হবে। তবে পুরোনো কোডকে নতুন এপ্লিকেশন এবং প্রোগ্রামে ফলাফল করার কাজে আপনাকে একটি বেশি সময় দিতে হবে এ ছাড়া উৎপীড়ক আর কোন সমস্যা নেই।

কোডটি করার বিয়তক যে আলোচনা এখন আমরা করবো তাদের মধ্যে রয়েছে ডিবেজ-৪ এর নতুন কনস্ট্যান্ট, রিড ইউটিলিটি ও ডিবি লিংক, কোডের পরিবর্তন বিষয়ে ইনস্ট্রাকশন নিয়ে সংক্ষেপে আলোচনা, এর ব্যবহার, ব্যবহারযোগ্য নতুন ক্লাস (Clause) এবং সর্বশেষে স্বয়ংক্রিয় হাণ্ডি ইউজার ক্লাসের (A.M.F.) এ ছাড়াও আলোচনার বিভিন্ন ধাপে এমিটিভরের ব্যবহার সংক্রান্ত কিছু প্রয়োজনীয় নীতিমালা এবং নেটওয়ার্ক ট্রান্সজেকশন প্রসেসিং নিয়েও বিস্তারিত ব্যাখ্যা দেয়া করবো।

কম্পাইলার : ডিবেজ ৩+ অপেক্ষা ডিবেজ ৪ এর যে পতি উন্নয়ন ঘটেছে তার মূল্য রয়েছে নতুন কম্পাইলারটি। এটি এক ধরনের টেকনোলজিয়ার যা একই সাথে প্রোগ্রামের এপ্লিকেশন স্পীড কমিয়ে দেয় এবং প্রোগ্রামের নির্মাণ হ্রাস করে। ডিবেজের নতুন এই ভার্সনটি যেকোনো সফল প্রোগ্রামেরই রান করার আগে স্বয়ংক্রিয়ভাবে কম্পাইল করে দেয়, তাই ডিবেজ-৩+ এর প্রোগ্রামগুলো ডিবেজ-৪ এ বেশ দ্রুতগতির রান করবে। Ashton-Tate দাবী করে যে, এই পারফরম্যান্স অস্বাভাবিক ১ হতে ১০ তন বৃদ্ধি পাবে। ব্যস্তের অর্ধেকই বা হলেও আমরা অন্ততঃ ৭ গুণ উন্নতি আশা করতে পারি।

প্রোগ্রামের সাথে আপনি যদি ডিবাগার (debugger) ব্যবহার না করেন তবে ডিবেজ-৪ কম্পাইল করা হারনি এমন কোন কোড রান করবে না। অবশ্য, কোন প্রোগ্রামকে কম্পাইল করতে হলে ডিট প্রপারটি হতে কেবল নিম্নোক্ত টেক্সটেরটি ব্যবহার করলেই চলবে :

DO Program-filename

আপনি ইচ্ছা করলে ডিবেজ-৪ সোড করার সময় ডন প্রপারটি থেকেও প্রোগ্রাম কম্পাইল করতে পারেন নিম্নোক্ত কমান্ড-লাইনের সাহায্যে :

dBase program-filename

উক্ত কোডেই ডিবেজ-৪ আপনার PRG ফাইলকে কম্পাইল করে DBO (DataBase Object) এপ্লিকেশন নিয়ে একই নামে আলাদা একটি ফাইল তৈরি করবে। কিন্তু পরবর্তিতে Compile কমান্ডটি ব্যবহার করে কোন ডিট প্রপারটি হতে কম্পাইল করা যায়। এটির Syntax হচ্ছে : Compile Program আন্যকোমেন্টেও মনে রাখতে হবে যে, ডিবেজ-৪ তখন কোন প্রোগ্রামের ডিটর DO program-filename-এই কমান্ডটি পাবে তখন ঐ প্রোগ্রাম ফাইলটিতে যে সর্বসময় প্রোগ্রামটির সাথে কম্পাইল করে ফেলবে।

আবার মনে করুন, আপনি একটি প্রোগ্রাম ফাইলকে কম্পাইল করছেন কিন্তু পরবর্তিতে ঐ ফাইলটিতে সাহায্য পরিবর্তন করছেন, অর্থাৎ ফাইলটি পুনরায় কম্পাইল করতে ভুলে গেছেন। এমন যদি আপনি DO কমান্ডটি ডিট প্রপারটি ব্যবহার করে প্রোগ্রামটি রান করেন তবে কিছু প্রোগ্রামটির পুরোনো DBO ফাইলটি রান হবে।

এই জাতীয় সমস্যা (বা ভুল) এড়ানোর জন্য আপনি ডিবেজ-৪ কে এমনভাবে সেট-আপ করতে নিতে পারেন যেন PRG এবং DBO ফাইলের ডস-ডেট এন্ড ডস-টাইম পরস্পর ভিন্ন হলে PRG ফাইলটি পুনরায় কম্পাইল করা হয় স্বয়ংক্রিয়ভাবে। এ কার্যের জন্য ডিবেজ-৪ এ প্রবেশ করলেই SET DEVELOPMENT ON এই কমান্ডটি ডিট প্রপারটি হতে দিতে হবে। কিংবা আরো ভাল হয় যদি কোনে ট্রাউট এন্ট্রির নিয়ে CONFIG.DB ফাইলের ৩৩তম SET DEVELOPMENT = ON লাইনটি লিখে দেওয়া যায়। ডেভেলপমেন্ট সেটিংকে এভাবে 'সব সময়' অন করা করার পক্ষে মনে লেগার কারণ হচ্ছে, যদিও এর ফলে এপ্লিকেশন স্পীড কিছু মাত্রায় হ্রাস পায় তথাপি, PRG ফাইলের পরিবর্তনের সাথে সাথে DBO ফাইলকেও আপডেট করার জন্য এটিই সুস্থর ব্যবস্থা।

কিড ইউটিলিটি এবং ডিবি লিংক : কম্পাইল করা কোড থেকে কোন DO কমান্ড নিয়ে যায় না করে আরও কার্যকরীভাবে ব্যবহার করার সহজ উপায় হচ্ছে একটি এপ্লিকেশনের অন্তর্গত সমস্ত সূত্র DBO কে একত্রিত করে একটি মাত্র EXE ফাইলে পরিণত করে তা ব্যবহার করা। ডিবেজ-৪ (Developer's Edition)-এ কার্যের জন্য BUILD নামক একটি ইউটিলিটি রয়েছে। ইউটিলিটিটির সাথে অন্যান্য যে প্রোগ্রামি টুল পাওয়া যায় তারা হবে DBLINK-নামক লিঙ্কার, কিছু টেমপ্লেট ল্যাংগুয়েজ কোর্স কোড ফাইল এবং টেমপ্লেট ল্যাংগুয়েজটির কম্পাইলার ও ইউটারপ্রাইর। কিছু LAN-বী জিডি ছাড়াও বিনা লাইসেন্স ক্রীতে রয়েছে অসীম রান টাইম সাপোর্ট। অবশ্য, এ কলামে আলোচিত অধিকাংশ কনভারশন টুলগুলোর জন্য অধিক ব্যাচহেল ডেভেলপমেন্ট এপ্লিশন ডিবেজের কোন দরকার নেই।

যা হোক, DBLINK এবং BUILD লিঙ্কার সূত্র ডিট ধরনের ট্রিগ্গারবাহী কাজ করে ; সমস্ত DBO ফাইলকে একটি মাত্র EXE-তে পরিণত করে, এবং ডিভিগারটি হচ্ছে এটি সিস্টেমকিড রিকম্পাইলেশন (SRC) তে সাহায্য করে। DBLINK ব্যবহারের Syntax বেশ সহজ :

DBLINK main-program-filename [/L]

main . . . . . এর স্থানে এপ্লিকেশনের প্রধান প্রোগ্রামটির নাম দিতে হবে। DBLINK তখন সমস্ত প্রোগ্রামটি প্রবেশ করবে অর্থাৎ প্রোগ্রামটি হতে সফল করা সমস্ত সিসিউটিস, সব প্রোগ্রাম বা প্রোগ্রাম ফাইল মূল প্রোগ্রামটির সাথে সংযুক্ত করে দেবে। /L অপশন (সুইচ) দ্বারা ক্রীনা অর্থাৎ ডিট কিংবা ফাইলটি-ডিভিগার করা যায়। যে ডিট ফাইলটি তৈরি হবে তার এক্সটেনশন হবে TXT, অপর দিকে এক সেট প্রোগ্রাম ফাইলের স্বয়ংক্রিয় নিঃশ্রেণ, কম্পাইল এবং লিঙ্ক করার জন্য মেনুডিরিক্ট একটি প্রোগ্রাম হচ্ছে BUILD. দুটি প্রধান পপ-আপ মেনু রয়েছে এই ইউটিলিটিতে। প্রধানটির নাম 'BUILD'। এর আরও ৪টি অপশন রয়েছে। Compile অপশন (১ম টা) দ্বারা যে প্রোগ্রাম ফাইল কম্পাইল করতে হবে তার নাম BUILD কে সরাসরি করা হয়। ডিভিগার অপশনটির নাম LINK। প্রোগ্রামটি কে কেবল কম্পাইল করতে হবে, না লিঙ্ক করার প্রয়োজন তা YES /NO দ্বারা সিঙ্গেল করা যায় এই অপশনটিতে। তৃতীয় অপশনটি হচ্ছে Output destination; যা দ্বারা DBO ফাইলকে প্রস্তুতকৃত অন্যান্য ফাইল অর্থাৎ ডেভেলপারী ও ড্রাইভে পাঠানো যায়। মেনুস সর্বশেষ অপশন হচ্ছে Perform BUILD। এটি একবারে তখনই সিঙ্গেল করবেন যখন Options-মেনু সহ এই মেনুটির সফল অপারেশন সফল প্যারামিটার সফীকরণে নির্ধারণ করে দেওয়া শেষ হবে। কেননা এপ্লিকেশন লিঙ্ক করার এটিই দ্রুততম অপশন।

দ্বিতীয় অর্থাৎ Options-মেনুতে সর্বশেষে ৬টি অপশন রয়েছে যার দুইটি প্রপে বিতৃত। তবে এর মধ্যে আমরা কেবল দ্বিতীয় অপশন 'Search for new functions' ও ড্রাইভে পাঠানো করব। বাকি ৬টি অপশনের মধ্যে ১ম এবং ৫ম অপশন দুটো হচ্ছেই বেগুনজ এবং বার্লি ৪টি অপশনের ডিটল মানেই সর্ব সাধারণের জন্য ব্যবহারযোগ্য। দ্বিতীয় অপশনটির YES প্যারামিটার দ্বারা প্রোগ্রাম এমন সমস্ত UDF টুলে ফেলে দেয়া যায় যাদের নাম ফুলডমে ডিবেজ-৪ এর অভ্যন্তরীণ কোন ফাংশনের নামে রাখা হয়েছে। BUILD-তখন এই কপ প্রতিটি UDF এর জন্য আলাদা এক মেসেজ লিখাবে। অন্যতরভাবে, আপনি যদি UDF তুলে সিঙ্গেল বাইপার করতে চান তবে তাদের নাম পাল্টে দিতে হবে। তবে আপনি যদি নিশ্চিত হন যে, এ ধরনের কৃত আপনাদের প্রোগ্রামে নেই তবে এই অপশনটির No প্যারামিটার সিঙ্গেল করলে কম্পাইলই স্পীড বৃদ্ধি পাবে।

কম্পাইলার ও লিঙ্কার বিষয়ে একত্রেই আলোচনার আমরা কোন ডিবেজ-৪ এ তাদের ব্যবহার নিয়ে অসহকোপাত করলাম। পরিকল্পনা দ্বারা এ দুটোর জন্য পরিচিত হন তারা '৯৪ জানুয়ারী সংখ্যার কমপিউটার জগৎ' এ 'ডাটাবেজ ল্যাংগুয়েজ হিসাবে ক্রীপার' শীর্ষক প্রবন্ধের হতবাহ পড়তে নিতে পারেন। ডাটাবেজ গ্রাফিক্স ব্যবহারকারীদের মধ্যে রাগ উঠতে হবে, প্রোগ্রাম এপ্লিকেশন স্পীড বাড়ানোর জন্য সহজলভ্য টুলগুলোর মধ্যে কম্পাইলার এবং লিঙ্কারই সবচেয়ে সর্বোৎকৃষ্ট সফলত্বপূর্ণ। এবং প্যাট্রাটোজি টেক্সট-এ 'কম্পাইলারের পারফরম্যান্স পরীক্ষা করার জন্য পরিচালিত বেসমার্চ টেস্টের একটি অংশ)

অর্ডারজটির অনেক কম্পিউটার মাধ্যমিক ডিবেক-৪ এর কপিআইজারকে অনেক টাই হুান দিয়েছে এবং এর শিফটও পঠিতও বেশ দ্রুত এবং সহজ।

কোড পরিবর্তনের করা বলাগেও এ পর্যন্ত আমরা সমস্যাও এ বিষয়ে আলোচনা করিনি। তবে আসুন, ইন্ডেক্স ফাইলের মাধ্যমে বিখ্যাতিকে কিছু অপেক্ষাপাত করা যাক।

**কোড পরিবর্তন :** MDX ফাইল নিয়ে আমাদের পূর্বের আলোচনার মাধ্যমে অল্পটা ত্রুটি কিংবা পরিষ্কার হওয়া উচিত যে, পুরোনো MDX ফাইলগুলো নতুন MDX ফর্ম্যাট বনাকার করার মাধ্যমে ডাটাবেজ কার্যক্রমে উন্নতি ঘটবে। এইরূপ না থাকার কোন কারণ নেই। তবে পুরোনো প্রোগ্রামগুলো যে অ্যপ্লিকেশন MDX ফাইলে প্রসেস করে তাদের অবশ্যই পরিবর্তন করতে হবে।

আরও মনে রাখবেন যে, DO ..... End Scan কম্পিউটারটির সমগ্র Scan While স্ট্রাকচার হতে অনেক দ্রুত। তাই যাবতীয় DO While কোডগুলো পরিবর্তন লাগবে অন্য প্রকাশ প্রবী। ডিবেক ৩+ হচ্ছে ডিবেক ৩ এ প্রোগ্রাম স্থানান্তরের সময় আপনাকে যদি সমস্যাটির পড়তে হয় এবং আপনার প্রোগ্রামের অনেক জায়গায় যদি DO while নিয়ে ডাটাবেজ এবং নুপে পড়েটার সূচ করলে হয় তবে যাবতীয় DO while এবং Skip পরিবর্তন করা হবে এক সূচ ধরনের বিকল্প।

পূর্বে আমরা এও উল্লেখ করেছি যে, Set skip কমাতে নিয়ে একের সাথে অনেক ডাটাবেজের সফল স্থাপন করা হয়। ডিবেক ৩+ এ এই ধরনের রিলেভেন স্থাপন নিয়ে অনেক দুর্লভতার রয়েছে। তাই প্রোগ্রামারদের নিম্নোক্ত এ জন্য সঠিক নিষেধ হয় এবং প্রোগ্রামটির ব্যক্তি কোডগুলো যথা দ্বারা হলে, ডিবেক-৪ এ Set skip এই সমস্যা সমাধান করে। তাই যদি ডাটাবেজ রিলেভেন হেরি করা হয়েছে যেখানে, পুরাতন প্রোগ্রামের সে সব স্থানে এটি ব্যবহার করা উচিত।

এরই ডিবেক-৪ এ আরো একটি নতুন এবং অতি প্রয়োজনীয় সমাধান নুই ডাইনামিকের এর। এতে এপ্লিকেটগুলোকে রে-কমান্ড ভিত্তিকে রেকোমেন্ড করা হয়। সেমেন নিচের উদাহরণের DECLEAR কমান্ডটি ঘারা একটি 3 X 16 এর টেবিল করা হয়েছে ঘার নাম ATEST, দ্বিতীয় লাইনে একটি [2, 12] এপ্লিকেট 503567 এই নিউমেরিক ডাটা স্টোর করা হয়েছে এবং (3,5) এপ্লিকেট "COMPUTER JAGAT" লেখাটি যাক হয়েছে।

```

Ex : 1
DECLARE ATEST (3,16)
ATEST [2,12] = 503567
ATEST [3,5] = "COMPUTER JAGAT"
উদাহরণে আপনি নিচের দেখতে পাচ্ছেন একটি এরের বিভিন্ন এপ্লিকেট ডাটা-টাইপ সেখানে যায় (এখানে নিউমেরিক এবং ক্যারেক্টার)। এর অর্থ হচ্ছে এদের সাহায্যে আপনি ইচ্ছা করলে COPY TO ARRAY এবং APPEND FROM ARRAY-এই দুইটি নতুন কমান্ড ব্যবহার করে দুইটি ডাটাবেজের মধ্যে কেরেক্ট আদান-প্রদান করতে পারেন। COPY , ... কমান্ডটি ব্যবহারের জন্য যে এরের ব্যবহৃত হার তা হয় 3 ডাইনামিকের এবং এতে ততটি এপ্লিকেট থাকতে হবে ডাটাবেজের যতটি ফিল্ড হয়েছে। নিচের উদাহরণটি দেখুনঃ

```

```

Ex : 2
DECLARE NAMES [16] && CUS-
TOMER has 16 fields

```

```

SELECT 1
USE CUSTOMER
COPY TO ARRAY names FOR CITY =
'Dhaka'
SELECT 2
USE CUSTTMP
APPEND FROM ARRAY names
Ex-3
SELECT 1
USE CUSTOMER
COPY TO TMP FOR CITY = 'Dhaka'
SELECT 2
USE CUSTTMP
APPEND FROM TMP

```

উপরের উদাহরণ দুটিতে দেখা যাচ্ছে একই কাজের জন্য ২ নং উদাহরণ ১টি বেশি লাইনের প্রোগ্রাম রয়েছে। অথবা এ উদাহরণটির পেশি-দ্রুততম। আসলে এখানে দুই সমস্যা হচ্ছে ৩+ নং উদাহরণের TMP ডিক ফাইলটি নিয়ে। ডিবেক ৩+ এর যে প্রসিডিউর দুইটি ফাইলটিতে কাজ করবে তাদের একবার শিখতে হবে এবং একবার পড়তে হবে। কিন্তু, এর থেকেই মেনে রাখতে অবশ্যই করা একটি জটিলতা তাই তাদের মাধ্যমে ডাটা আদান প্রদান হলে ট্রান্সফার শীত অনেক দ্রুত হবে। কারণটি কি আশা করি পরিষ্কার ভা জানেন। এছাড়াও COPY TO ARRAY কমান্ডটিতে অপশন হিসাবে একটি ফিল্ড নিউ প্রোগ্রামটির সরকরে করা যায়। যে এরের সর্বস্বত্ব করা হয় তা যদি নুই ডাইনামিকের হয় তবে প্রথমটি ঘারা কতটি রেকর্ড এবং দ্বিতীয় এপ্লিকেট ঘারা কতটি ফিল্ড ট্রান্সফার করতে হবে তা নির্দিষ্ট হয়।

যা ব্যবহারযোগ্য নতুন কমান্ড। USE কমান্ডটিতে এমন নতুন কিছু ক্রম যোগ করা হয়েছে যাদের মাধ্যমে কোন ডাটাবেজকে আন-সিলেক্টেড ওয়ার্ক এরিয়াতে খোলা যায়। এই ডাটাবেজগুলো একের অধিক ওয়ার্ক এরিয়াতে খোলা যায় এবং কোন ডাটাবেজের খোলা এমন ভাবে সরক করা যায় যাকে ডাটাবেজটি কেলে READ only থাকে। অর্থাৎ এতে যে কোন পরিবর্তন রোধ করা যায়। ডাটাবেজকে আনসিলেক্টেড ওয়ার্ক এরিয়াতে খোলার জন্য নিচের Syntax ব্যবহার করা যায়।

```

USE filename IN work area number
এইরূপ কমান্ডের মাধ্যমে বর্তমানে সিলেক্ট করা ওয়ার্ক এরিয়াতে কোন পরিবর্তন হয় না।

```

USE এর সাথে ব্যবহারের জন্য নতুন যে ক্রমগুলো যোগ করা হয়েছে তাদের মধ্যে AGAIN-কী ওয়ার্কটি একটি অসহাজবিক। এই কী-ওয়ার্কটি ঘারা একই ডাটাবেজকে দুইটি ভিন্ন ওয়ার্ক এরিয়াতে খোলা যায়। অর্থাৎ একই ডাটাবেজের জন্য তখন দুইটি আদান প্রসিডিউর বা প্রাইমারী ইন্ডেক্স ব্যবহার করা যায়। SEEK এবং FIND কাজে এক্ষেত্রে এর সুবিধা হচ্ছে প্রাইমারী কী-টার কোন পরিবর্তন ছাড়াই দুই টি আদান প্রসিডিউর করা সম্ভব। কি ডাটা AGAIN প্রসিডিউর ব্যবহার করতে হবে তা নিচের উদাহরণ দেখুন।

```

SELECT
USE customer AGAIN ORDER custcode
USE customer AGAIN IN 2 ORDER zip
এমন আদান 1নং ওয়ার্ক এরিয়ায় custcode এবং ১ নং ওয়ার্ক এরিয়ায় zip ফিল্ড দুটোতে সিলেক্ট প্রোগ্রাম-লগিক ব্যবহার করে FIND এবং SEEK চলাতে পারেন-
FIND <custcode value>

```

```

DISPLAY custcode, zip
SELECT 2
FIND <zip code value>
DISPLAY cust code, zip
ফলে ক্রীনে আদান দুটো ওয়ার্ক এরিয়ায় বিভিন্ন রেকর্ড দেখা হবে। আসলে উপস্থান দেখতে পারেন। তবে Ashton-Tate এর মতে AGAIN কী ওয়ার্কটি ডাইনামিক সমস্যার সূত্র করতে পারে। বিশেষতঃ যদি দুটা ওয়ার্ক এরিয়াই একটি ইন্ডেক্স ব্যবহার করে। এ অবস্থায় ডাটা সমাধান বা এন্ট্রিতে ফলে ইন্ডেক্সটি নী (corrupted) হয়ে যেতে পারে। তাই আমাদের পরামর্শ হবে- ডাটা এন্ট্রিতে সমা USE- এর সাথে AGAIN ব্যবহার না করাই উত্তম।

```

সংশোধন USE filename .NOUPDATE ঘারা ডাটাবেজ কোন পরিবর্তন বা পরিবর্তন বন্ধ করে দেওয়া যায়। পূর্বের বিভিন্ন ফীচারগুলোর মত এই নতুন ক্রমগুলো ব্যবহারের সূত্র মাধ্যমে ডিবেক ৩+ অপেক্ষা অধিক দ্রুততার প্রোগ্রাম লেখা সম্ভব বা হজরতই এই প্রোগ্রাম শীত ব্যক্তিগত দেখে।

হার্ডওয়্যার ইউজার ফীচার : আপনি যদি নেটওয়ার্ক প্রসিডিউর কাজ করলে তবে ডিবেক-৪ এর AMUF (Automated multiple user Features) ডিবেক ৩+ এর অনেক কোডের স্থানে ব্যবহার করা সম্ভব। এটি ডিবেক-৪ এ প্রচলিত অনুসারে ব্যবহারকারের রেকর্ড বা ডাটাবেজ লক করে ফেলেবে। এছাড়াও হার্ডওয়্যার রি-ট্রাই এবং টাইম-আউট ও সার্ভিসের মাধ্যমে ডাটা হারিয়ে এবং নেটওয়ার্ক এন্ট্রিতে জন্য রিমেট টাইম-আউট (RTSU)-ও নতুন ডার্নালিতে পাওয়া যাবে।

তাই ফাইল লকিং এর যথেষ্ট রি-ট্রাই ইমপ্লিমেন্টের জন্য আপনাকে পূর্বে যে প্রসিডিউর লিখতে হয়েছিল এখন তার আর প্রয়োজন হয় না। পরিবর্তন করবে এই কমান্ড একটি টেকের সফলভাবে খবর চোঁটা খোলাতে কাজ করে করতে পারে। যেমন EDIT 3 ঘারা ৩নং রেকর্ডটি লক করে পড়বে। অথবা যে সকল কমান্ড ডাটাবেজ থেকে রেকর্ড পরিবর্তন করে তাদের ব্যবহার সম্ভব ডাটাবেজটি লক হয়ে পড়বে। উদাহরণ রূপ REPLACE ALL CUST CODE WITH "00" ঘারা সম্পূর্ণ ফাইলটি লক করা হবে।

এই শব্দই কমান্ডগুলোর একটি অসুবিধা হচ্ছে এদের ডাটাবেজের আয়তন শুধি করে দেয়। সেসর ব্যবহারের পূর্বে কোন এপ্লিকেশনে মিডোল কমান্ড লাইন দুইটি লিখতে নিতে হবে-

```

USE database name
CONVERT
CONVERT কমান্ডটি ডাটাবেজের একটি 16 বাইটের ফিল্ড যোগ করে। এই ফিল্ডটিতে যে রেকর্ড ৩ ঘা ফাইল লক করা হয়েছে তা সম্পর্কে তারিখ ও সময় ভিত্তিক কিছু তথ্য, লক করা অবস্থায় নেটওয়ার্ক নামের পূর্বক আউট এবং কোন ব্যবহারকারী এটি লক করেছে তার নাম সর্বাধিক হয়। এই ফিল্ডটির নাম DBASELOCK. আপনি যদি নেটওয়ার্কের মাধ্যমে বেশি বা কম অংশ সর্বেক্ষণ করতে চান তবে CONVERT n এর কমান্ডটি ব্যবহার করতে পারেন। যেখানে n এর মান ১ হতে 16 এর ভিতর থাকতে হবে। n হচ্ছে DBASELOCK ফিল্ডের জন্য কত বাইট খায়েগে অ্যুয়েনালিট হবে তার নাম। নেটওয়ার্কের নামের জন্য যে ক্লাসপা লাগবে তা n-ও সূত্রের মাধ্যমে বের করা হবে। এই ডিবেকের AMUF এর জন্য এই ব্যক্তিগত ডিক

```

শেদের সুবিধাও আছে। এর সাহায্যে রেকর্ড এবং ডাটাবেজ লকিং সম্পূর্ণ অটোমেটিক এবং স্রেষ্ঠতম হয়। তাছাড়া (যদিও আমরা এখানে বিস্তারিত আলোচনা করছি না) আপনি এই DBASELOCK ফিচারে প্রবেশ করে নানাবিধ গুরুত্বপূর্ণ তথ্য সম্বন্ধ রচনাতে পারেন।

**ডিবেক্স-৪** এর মাটিট ইন্টারফেস ফীচারের সর্বাপেক্ষা আশ্চর্যকর ব্যাপার হচ্ছে এতে রিয়েলটাইমের ডিভিডেড জীন আপডেট সাহিত্য হয়। যেমন কোন ব্যবহারকারী যদি একটি রেকর্ডের পরিবর্তন করতে থাকে তবে এই রেকর্ডটি দেখছে এমন সকল ব্যবহারকারী প্রায় সাথে সাথে তাদের জীনে রেকর্ডের বিপেণ্ট পাবে। রেকর্ড আপডেট এবং জীন আপডেটের মধ্যে কত সময় দেবী হবে তা SET REFRESH TO n কমান্ড দ্বারা পূর্ণ নির্ধারণ করে দেওয়া যায়। এখানে n হচ্ছে সেকেন্ড নির্দেশক একটি মিনিটেরিক ভেসু। আপনি প্রথম n এর মান 5 দিয়ে শুরু করতে পারেন। পরে নেটওয়ার্ক পারফরমেন্স লক্ষ্য করে একে বাড়াতে বা কমাতে পারেন। তবে n এর মান খুব কম মিলে নেটওয়ার্ককে ঘন দান জীন আপডেট করতে হবে, ফলে অন্যান্য কার্যক্রমগুলো প্রবেশ করতে দেরি হবে।

**প্রিন্টিংর কোড** ডিবেক্স-৪এ প্রিন্টিংয়ের জন্য আলাদা ফাইল তৈরি করতে হলে এ ধরনের কোন বাধ্যবাধকতা নেই। প্রোগ্রামের শেষে প্রিন্টিংর কোড রাখা যায়। অর্থাৎ যে প্রিন্টিংগুলো বিভিন্ন প্রোগ্রামে ব্যবহার করে কেবল সেগুলোকেই আলাদা ফাইলে রাখতে হয়, যদি প্রিন্টিংর কোড মাত্র একটি মডিউল ব্যবহার করে তবে তা এই মডিউলের অন্তর্ভুক্ত করে রেখেই সম্ভব।

**ডিবেক্স-৪** একটি সময়ে ১,১৭০টি প্রিন্টিংর একটির করা যায়। যার ফলে ট্রান্সকার প্রোগ্রামিংএ প্রিন্টিংয়ের ব্যবহার নিয়ে কোন সমস্যা হওয়ার কথা নয়। তাছাড়া প্রিন্টিংয়ের হ্রাসে অধিক নফিস্ট্যান্ট UDF ব্যবহার করা যায়। এইরূপ একটি UDF (এবং UDF-টি কম করার নতুন) দ্বারা কীবোর্ডের ক্যাণ্টের ট্রিগ হতে ডান এবং বাম উভয় দিকের অক্সিজেনালিসের প্রিন্স (CHR : 10) সরাসরে যায় তা দেখানো হলে। প্রোগ্রাম ব্যবহারকরকাজ সহজসাধ্য করা ছাড়াও যদি প্রিন্টিংর এবং ফাংশনের সঠিকভাবে ব্যবহার করা যায় তবে তারা প্রোগ্রাম শীঘ্র ও বৃদ্ধিতে সাহায্য করে।

```
*Program MAIN.PRG
* Function LR.TRIM ( ) is user-defined
SCAN
lc_pict = lc_pict + ALLTRIM (Last name) + ' '
ENDSCAN
*EOP : MAIN.PRG
*Function ALLTRIM ( ) removes leading
* and trailing blanks of a string
FUNCTION alltrim
PARAMETERS lc_string
PRIVATE ret_string
ret_string = LTRIM(RTRIM(lc_string))
RETURN (ret_string)
```

**ডিবেক্স-৪** এ ট্রান্সকেশন প্রসেসিং : ডাটাবেজ ইন্টিগ্রেটি (নিরাপত্তা) রক্ষার ক্ষেত্রে ট্রান্সকেশন প্রসেসিং (TP) এর ধারণা উন্নত টেকনিক। নেটওয়ার্ক বা মাল্টিইউজার পরিবেশে এটি অলি-অবশ্যক, কেননা সেখানে যে কোন সময় অনেক ব্যবহারকারীদের কেউ

ডাটাবেজে পরিবর্তন করতে পারে। একক ব্যবহারকারীর পরিবেশে TP দ্বারা সাধারণ কিছু বিচার হতে ফাইলের ডাটা রক্ষা (যেমন ওয়ার্ড পারফরমেন্স টাইমড থ্যাঙ্ক করা), রেকর্ড বা ফাইল লক এবং অন্যান্য নানা মেমব্রী সক্রিয় সমস্যা ডাটাবেজকে নষ্ট হয়ে যাওয়ার হাত থেকে রক্ষা করতে বুঝায়। এটিও একধরনের মিরর ইমেজ প্রসেসিং, যার সাহায্যে নষ্ট হয়ে যাওয়া ডাটাবেজ ফাইলকে রিট্রিভ বা রিকন্সট্রাক্ট করা যায় তার মূল্যবান ডাটামনে। ট্রান্সকেশনের কিছু বিষয় নিয়ে আমরা এখন সর্বাধিক আলোচনা করব, কেননা অনেক ব্যবহারকারীর কাছেই এটি ভেমন পরিচিত নয়।

সরেক্ষেপে ডিবেক্স-৪ এর তাথায় ট্রান্সকেশন কিছু সম্ভাব্যতার কার্যধারা—এমন কিছু ডাটা আপডেশন বা একটি বা অধিক ডাটাবেজকে মডিফাই করে। সাধারণত এট্রিকেশন প্রোগ্রামারই ট্রিক করে যেন এইরূপ একটি ট্রান্সকেশন প্রোগ্রামের কয়টি ডাটাবেজ কমান্ড সম্বন্ধে হয়। অর্থাৎ ট্রান্সকেশনের সমস্যাগুলো কতটুকু গীর্ণ হবে তা আপনাকে নির্ধারণ করে দিতে হবে। ট্রান্সকেশনের পূর্বে ডাটাবেজের যেরূপ স্থিতিশীল অবস্থায় ছিল ট্রান্সকেশন ঘটান পরও ডাটাবেজ সেইরূপ অবস্থায়—এ ধরনের নিশ্চয়তা TP প্রদান করে। তবে ভাই বলে এই নয় যে কোন TP-পরিষদ চলাকালে ডাটাবেজের পরিবর্তন করা হলে তা সরেক্ষেপ করা হবে না। আপনাকে TP-এর মূল কমান্ডসই হচ্ছে একটি এরূপ ট্রান্সকেশন চলাকালে ডাটাবেজকে একটি অ-স্থিতিশীল অবস্থায় রাখতে দেওয়া হয়। এখন ট্রান্সকেশনের কোন কমান্ড চলাকালে যদি কোনরূপ বাধা দান করা হয় তবে ডাটাবেজকে পূর্বে স্থায় অবস্থায় ফিরিয়ে আনা হবে। অন্যথায় ডাটাবেজের সমস্ত এন্ট্রি, ডিফিট বা পরিবর্তন ছাড়াবিকভাবে গ্রহণ করা হবে।

ট্রান্সকেশন খণ্ড দেওয়ার প্রেক্ষিতে ডাটাবেজকে পূর্বাভাস ফিরিয়ে আনার প্রক্রিয়াতে ক্লাই বক সেল-ব্যাক (Roll back)। একে এইরূপ বলার কারণ বাধা দেওয়ার অর্থ পর্যন্ত সমস্ত কমান্ডের টেস্টটি তখন ঘটে। যেমন ডিফিট করা হলে হবে ইনসার্ট ক্লাই, এন্ট্রি করা হলে হবে আন-কু-এর কাজ-ই-ভাটি। এরূপ ভাবে, ডাটাবেজকে এমনভাবে তার প্রকৃত অবস্থায় এনে দেওয়া হবে, যে মনে হবে ট্রান্সকেশন নামক কিছু ঘটেনি (যেমন হিসাবরক্ষণ লেন-দেনের তালিকা তৈরি-ক্রেরিট সম্বন্ধে হলে ব্যালেন্স দেবে হয়।। শূন্য তেমনই ট্রান্সকেশন প্রসেসিং-এ ডাটাবেজকে যোগ-বিয়োগ করে পূর্বাভাস ফিরিয়ে আনা হয়।

ট্রান্সকেশন পরিষদ : ডিবেক্স-৪ বেশ কিছু কমান্ডের মাধ্যমে ট্রান্সকেশন প্রসেসিং-এ পূর্ণ পিয়ার প্রদান করেছে। এই কমান্ডগুলো এট্রিকেশনের মধ্যে বা ডাট প্রস্পট-টুই ইমেজ হতেই ব্যবহার করা যায়। এ কমান্ডগুলোর যথোপযথ্য হচ্ছে BEGIN TRANS-ACION, কমান্ডটির মাধ্যমে কোন ট্রান্সকেশন প্রসেস শুরু করা হয়। এই কাজ ততক্ষণ পর্যন্ত চলে ততক্ষণ END TRANSATION কমান্ডটি দেয়া হলে না হয়; অর্থাৎ এই দুইটি কমান্ডের মধ্যবর্তী সময়ই হচ্ছে ট্রান্সকেশন পরিষদ। তবে এই কার্যক্রম মাত্রাতিরিক্ত সময়সীমাপন যেন না হয় সে নিকে মুক্তি রাখতে হবে। কেননা ট্রান্সকেশন পরিষদ চলাকালীন সমস্ত রেকর্ড এবং ফাইল লক হয়ে যায়।

ডাটাবেজের প্রতিটি ট্রান্সকেশন এবং এর সাথে জড়িত সমস্ত পরিবর্তন লগ ডিবেক্স-৪ একটি ট্রান্সকেশন কলম তৈরির মাধ্যমে সরেক্ষণ করে END TRANS-ACION কমান্ডের পূর্ণ পর্যন্ত এটিও সংঘত থাকে। END TRANSATION এর মাধ্যমে ডিবেক্স-৪ বুকে

নেয় যে ট্রান্সকেশন কাজ সফলতার সাথে শেষ হয়েছে। সে তখন লগ ফাইলটি এবং সমস্ত লক করা ডাটাবেজ রিলিজ করে দেয়।

ট্রান্সকেশন চলাকালীন যদি কোন এর দেখা দেয় তবে ডিবেক্স-৪ ট্রান্সকেশন লগফাইলটি ব্যবহার করে ডাটাবেজকে আদি অবস্থায় নিয়ে আসে। ROLLBACK কমান্ডটির মাধ্যমে এ কাজটি করা যায়। একই ব্যবহারকারীর সিস্টেমে এই ট্রান্সকেশন লগ ফাইলটির নাম TRANSGE.LOG এবং কোন নেটওয়ার্ক স্টেশনে একে ডাকা হয় Workstation.LOG যেন। যেখানে workstation লগ যে নেটওয়ার্ক স্টেশনটিতে সিস্টেমে হতে BEGIN TRANSACTION কমান্ডটি ইঙ্গু করা হয়েছে। তার নাম কে বুঝায়। একটি NOVEL নেটওয়ার্ক এই স্টেশন নাম সাধারণ ব্যবহারকারীর লগ ইন JD হয়ে থাকে।

**ডিবেক্স-৪** এর LKSYS ( ) ফাংশনটি দ্বারা কোন ব্যবহারকারী এইরূপে ডাটাবেজ ও রেকর্ড লক এরের জন দায়ী তার নাম এবং কোন তারিখ ও সময়ে রেকর্ড বা ফাইলটি লক হয়েছে তা জানা যায়।

মাল্টিইউজার ফীচার নিয়ে আলোচনায় আমরা যে DBASELOCK ফিচারটি নিয়ে আলোচনা করছি LKSYS ( ) ফাংশনটি এই ফিচার ব্যবহার করেই এই সব তথ্য প্রদান করে।

ডিবেক্স-৪ এর অধীনে ট্রান্সকেশন প্রসেসিং ব্যবহার করতে চাইলে প্রত্যেক ব্যবহারকারীকে আলাদা নাম (USER ID) নিয়ে নেটওয়ার্ক লগ-ইন করতে হবে। কেননা ডিবেক্স-৪ একটি নামের বিভিন্ন ব্যক্তি কর্তৃক ফাইল লক-আপ বিষয়ে কোন ভারতম্য বুঝতে পারে না। এইরূপ ক্ষেত্রে LKSYS ( ) এর ফিচার ভেসু একই হবে।

ট্রান্সকেশন প্রসেসিং এর সাথে যুক্ত হয়ে এর প্রসেসিং এ সাহায্য করার জন্য ডিবেক্স-৪ বেশ কিছু কমান্ড এবং ফাংশন সরবরাহ করেছে। এদের মধ্যে ON ERROR কমান্ড দ্বারা একটি প্রোগ্রাম সরবরাহ করা হয় যাতে কোন এর দেখা দিলে প্রোগ্রামটি কন্ট্রোল লাভ করে। এর মাধ্যমে এর প্রসেসিং-এ বেশ কিছু ক্ষমতা লাভ করা যায়। যেমন মনে করুন, ফাইল লক করে রাখা হয়েছে—এ ডিভিডে আপনি ব্যবহারকারীকে সূচন জীন উইচারে মাধ্যমে একটি স্মিট্রিশ দেখাতে পারেন এবং তার পরবর্তী কার্যক্রম হিসাবে কিছু অপশন সরবরাহ করতে পারেন। যেমন RETRY কমান্ডটি দ্বারা "Retry" নামক অপশন নিতে পারেন যার মাধ্যমে ফাইল লক রিভুড করা হয়েছে কিনা তা বার বার চেক করে দেখা যায়।

ট্রান্সকেশন প্রসেসিং সক্রিয় এইরূপে আছে কিছু ফাংশন/কমান্ডের মধ্যে ROLLBACK ( ) ফাংশনটি ROLLBACK কমান্ডের সম্ভলতা ট্রিক করা করে ISMARKED ( ) ফাংশনটি চাফা কোন ট্রান্সকেশন প্রসেসিং চলছে কিনা তা চেক করা যায়। এবং পরিশোধে COMPLETED ( ) ফাংশনটি দ্বারা ট্রান্সকেশন প্রসেসিং ট্রিক করা সম্ভব হয়েছে কিনা তা জানা যায়। এটি কেবল চেক করে সম্ভলতা ও বিকলজ বিষয়ে, কিছু ON ERROR- এর ঘট রেজি জারীর কোন কার্যক্রমে সাহায্য করেন।

ট্রান্সকেশন প্রসেসিং-এর তুলনামূলক ভারতম্য। যদিও ট্রান্সকেশন প্রসেসিং ডিবেক্স-৪ এর অগ্রগত পারফরমেন্সের একটি, তথাপি এটি একটি পূর্ণ নিশ্চয়কর নয়। অনেক বিশেষণাল ডাটাবেজ সিস্টেমে ডাটাবেজের

পরিবর্তন ডাফফিকি বেকর্ড করা হয়। তবে এই পরিবর্তন সাধন করা হয় একটি আলাদা 'shadow' ডাটাবেজে, এর মাধ্যমে মনে হবে যেন আসলে ডাটাবেজটিই পরিবর্তিত হয়েছে; কিন্তু প্রকৃত পক্ষে তা নয়।

এইরূপ RDBMS (Relational Database Management System)-এ ব্যবহারকারী যদি তার পরিবর্তন নিয়ে সন্তুষ্ট হয় তবে সে COMMIT জাতীয় কোন কমান্ড দেনা এবং পরিবর্তনগুলো ডাটাবেজে লিপিবদ্ধ হয়। আর সে যদি সন্তুষ্ট না হয় তবে প্রথম হতে শুরু করতে পারে ROLLBACK কমান্ডটি দিয়ে। এ অবস্থায় ফুল ডাটাবেজটির সমগ্রসমগ্র (কেননা এতে কোন পরিবর্তন লেখা হয়নি) Shadow ডাটাবেজেরই পূর্বাবস্থায় ফিরিয়ে আনা হয়। এভাবে, কেবল এদের রিকভারীর জন্যই নয়, ডাটাবেজে ব্যবহারকারীদের সমস্ত পরিবর্তন যাতে রুটিন মাসিক রিটার্ন করা যায় সে অবস্থায় RDBMSগুলোতে রয়েছে।

ডিবেক্স-৪ এর উদ্দেশ্যটি ঘটে। ডাটাবেজের যাক্রমিক পরিবর্তন ভৎক্ষণাৎ আসল ডাটাবেজেই লিখে ফেলা হয় এবং এই পরিবর্তন পাশ্বে লিখে আপনাকে সিঙ্ক্রিয়াল রোলব্যাক শিকেরিয়ে দেতে হবে। সুতরাং এই দুইটি পদ্ধতির মধ্যে পার্থক্য হচ্ছে-ঐতিহ্যবাহী পূর্বোক্ত ট্রান্সলেকশন প্রসেসিং-এর ইমপ্লিমেন্টেশন হচ্ছে এপ্রিকেশন (যা কোন ক্ষেত্রে ব্যবহারকারী) কেন্দ্রীক। অর্থাৎ ব্যবহারকারী বা এপ্রিকেশনের লজিকই সিঙ্ক্রিয়াল গ্রহণ করতে কখন আসল ডাটাবেজে পরিবর্তন সাধিত হবে কিংবা কোন অবস্থায় রোল-ব্যাক করতে হবে।

অপরদিকে ডিবেক্স-৪ এর একই বিষয়ে ইমপ্লিমেন্টেশন হচ্ছে সম্পূর্ণ এর কেন্দ্রীক। অর্থাৎ ট্রান্সলেকশন সব সময়ই ডাটাবেজে বেকর্ড করা হবে তবে কোন এরর দেখা দিলে তা রোল-ব্যাক করা হবে।

যা হোক, ডিবেক্স-৪ এর এই সীমিত ট্রান্সলেকশন প্রসেসিং ক্ষমতা 'নাই মামার চেয়ে কানা আমার মত'। এবং ডিবেক্স ডেস্কোপারদের জন্য সুখবর হচ্ছে এপ্রিকেশনের এরর ডাটা কন্ট্রোল সিস্টেমের মাধ্যমে হাতের নাখালে পেরোচ্ছেন।

জানার আছে অনেক কিছু? ডিবেক্স-৪ এ এমন অনেক উপকারী ও চমকপ্রদ খতিয়ান রয়েছে যা প্রবন্ধের এই ছোট পরিসরে আলাচনা করে আমরা কুলিয়ে উঠতে পারিনি যেমন মেগার সিস্টেমের সাথে ব্যবহারের জন্য বর্তমান অনেক শক্তিশালী বেশ কিছু কাংশন রয়েছে যা ডিবেক্স-৩ এ অনুপস্থিত। এ সংশ্লিষ্ট লিখে আপনাকে কিছু চর্চা করতে পারুন। আরো একটি বিষয় নিয়ে অনুশীলন করতে ভুলবেন না যেন: তাহলে QBE (Query By Example)-এর সম্ভাবনা। এটি হচ্ছে মাস্টি ডাটা ফাইল রিপেশন স্থাপনের সহজ পথ। পরিশেষে, এটাও খেয়াল করে দেখবেন যে, CONFIG.DB ফাইল বর্তমানে ৪টি পর্যন্ত প্রিন্টার ইনস্টল করতে দেয় এবং প্রতিটি প্রিন্টার ড্রাইভারের জন্য ৪টি পর্যন্ত সফট ফন্ট সেজ করতে দেয়।

অবশেষে এটিই বলতে হয় যে, ডিবেক্স-৪ সর্বোৎকর্ষ উত্তম নয়। তবে এক জার্নাল হতে পরবর্তী জার্নালে লাফ দিয়ে একটা উন্নতি সাধন খুব কম সফটওয়্যারই দেখাতে পেরোচ্ছে। ডিবেক্স ৩+ হতে ডিবেক্স-৪ ও ফুনাকরণের মাধ্যমে নটীকৃত পরিবর্তন আপনাকে স্বস্তি দেবে। এই নতুন পরিবেশে এসে যদি নতুন দেখবেন তাহলে এই প্রবন্ধের 'কনজারেশন স্ট্রাটেজী' অনুসারে বিচার করুন যত্নের সাথে। উইকো এবং মেনু টৈরির ক্ষমতাকে পূর্ণাঙ্গায় সকল এপ্রিকেশনে ব্যবহার করুন, যান, ট্রান্সফার সিস্টেমের পরবর্তী কাজগুলো ডিবেক্স-৪ নিজেই করে নেবে।

REFERENCE FOR NEW USERS :

1. QUICK GUIDE TO DBASE / BY : David Kolodney.
2. THE ABC'S OF DBASE IV / BY : Robert Cowart.
3. Understanding DBASE IV BY : Alan Simpson.

FOR MID-LEVEL USERS AND EXPERTS :

1. MASTERING DBASE IV PROGRAMMING / BY : Carl Townsend.
2. DBASE IV USERS INSTANT REFERENCE / BY : Alan Simpson.
3. DBASE IV PROGRAMMERS INSTANT REFERENCE / BY : Alan Simpson.
4. DBASE IV USERS DESKTOP COMPANION / BY : Alan Simpson.
5. DBASE IV PROGRAMMER'S REFERENCE GUIDE / BY : Alan Simpson.

FOR ALL PERSONS :

1. INSIDE DBASE IV / BY : Tony Lima.

(মাস্টি)

# WE OFFER

## THE VALUABLE PCS AT THE LOWEST PRICE

SYSTEM	386SX
SPEED	40 MHz
RAM	2 MB
FLOPPY	3.5" HD
HARD DISK	210 MB
MONITOR	SVGA MONO
PRICE	Tk. 29,000/=

SYSTEM	386DX
SPEED	40 MHz
RAM	4 MB
FLOPPY	3.5" HD
HARD DISK	260 MB
MONITOR	SVGA MONO
PRICE	Tk. 35,000/=

SYSTEM	486 SLC2
SPEED	50 MHz
RAM	4 MB
FLOPPY	3.5" HD
HARD DISK	420 MB
MONITOR	SVGA MONO
PRICE	Tk. 41,000/=

SYSTEM	486DX2
SPEED	66 MHz
RAM	8 MB
FLOPPY	3.5" HD
HARD DISK	540 MB
MONITOR	SVGA MONO
PRICE	Tk. 60,000/=

For SVGA Color Monitor ( 0.39 mm )  
Add Tk. 6,000/=

FOR SERVICING, MAINTENANCE &  
TROUBLE-SHOOTING OF COMPUTER  
PLEASE CONTACT :

## BANGLADESH COMPUTER SERVICES

RAJANIGANDHA BHABAN  
74/1 LABORATORY ROAD, DHAKA 1205.  
(Near Coffee House)